Defense Information Infrastructure (DII)

Common Operating Environment (COE)

Programmer's Reference Manual

Version 3.0

31 October 1996

Prepared for:

Defense Information Systems Agency

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Attachment 1
       Application Programmer Interface (API) Reference Guide   Draft Version 3.0.0.
       (HP and Solaris)

This page intentionally left blank.

1. Introduction

1.1 Purpose

The *DII COE Programmer's Reference Manual* is intended to be a reference document for information about the Defense Information Infrastructure (DII) Common Operating Environment (COE). This manual has been created, in conjunction with the *DII COE Programmer's Manual*, to support programmers' development activity in the DII COE.

This manual provides specific information concerning the component structure and toolsets currently being used in the DII COE, and serves as a catalogue of published standards and COE related documentation. Part of the concept of this manual is to provide new COE users and programmers with a quick reference guide to applicable standards and specifications, including Application Programmer's Interface (API) documents, which control and shape system and application development within the DII COE architecture.

Except where indicated, the contents of this document apply to all currently supported COE platforms (UNIX and Windows NT). Information which applies to a single platform or operating system is identified with examples to denote the difference.

The Defense Information Systems Agency (DISA) will review and update this document as required to remain current with the evolution of the DII COE. This document supersedes Version 2.0, all earlier draft versions, presentations, or working group notes. Please direct any comments or questions regarding the *DII COE Programmer's Reference Manual* to:

Point of Contact (POC)	Cmdr. Charles B. Cameron
	Chief, DII COE Engineering Division
Address	DISA/JIEO/JEXF-OSF
	45335 Vintage Park Plaza
	Sterling, VA 20166-6701
Telephone	(703) 735-8825
Fax	(703) 735-8761
Internet Address	cameronc@ncr.disa.mil

1.2 Scope

This document describes the technical aspects of the DII COE, its major components, and the resources provided for COE development activities. It also describes the technical requirements for building and integrating software components on top of the DII COE. This document discusses the various segments for the current COE platforms, and the purpose and usage of each segment.

This document provides implementation details which describe, from the perspective of DII COE software development, the following:

- the standards and reference documents which govern the DII COE,
- the COE development process,
- the on-line resources which support COE development,
- the guidelines for COE development,
- the COE developers' toolkit,
- the Application Program Interfaces (APIs) supported in the current COE release,
- the components of the current COE release.

1.3   DII COE Overview

The DII COE is best described as an architecture that is fully compliant with the *Department of Defense, Technical Architecture Framework for Information Management* (TAFIM). The COE is a foundation for building an open system designed around a client/broker/server model. The DII COE encompasses architecture, standards, reusable software modules, and automated integration tools in a cohesive framework for systems development.

Each system built upon the COE foundation uses the same set of APIs to access common COE components, the same approach to integration, and the same set of tools for enforcing COE principles. Precisely the same COE software components are used for common functions, such as communications interfaces and dataflow management.

1.4   Document Structure

This document is divided into 5 major sections.

Section 1   discusses the purpose and scope of this document. It also contains an overview of the DII COE.

Section 2   discusses standards and reference documentation.

Section 3   gives an overview of the DII COE development process.

Section 4   describes DII COE components for UNIS systems.

Section 5   describes DII COE components for Windows NT.

Appendices   There are three appendices (A through C) as follows:

      A.   List of Acronyms
      B.   List of COE Tools & APIs
      C.   List of POSIX Function Calls (APIs)

Attachment 1 Application Prograzmmer Interface (API
              Reference Guide(Version 3.0.0.3 for HP and Solaris)

This page intentionally left blank

2.   Standards and Reference Documents

This section will cover the COE environment in terms of applicable standards. Appendix E of the *DII COE Programmer's Manual* covers the list of Adopted Information Technology Standards of the TAFIM.

The DII COE reflects the principles, architecture, and standards adopted by TAFIM which includes the Portable Common Toolkit Environment (PCTE) and the ongoing work in the Integrated Software Engineering Environment (ISEE).

2.1   Reference Documentation

Refer to Appendix C of the *DII COE Programmer's Manual* for a complete list of reference documents.

POSIX.1 is the Portable Operating System Interface that is currently a Federal Information Processing Standard (FIPS) 151-2 based on ISO/IEC 9945-1 international standards and born from IEEE POSC P1003.1 American National Standards Institute (ANSI) Standard. POSIX.2 , is part 2 of the Shell and Utilities is currently the FIPS 189 based on the ISO/IEC 9945-2 international standard and born from IEEE POSC P1003.2 (ANSI) Standard.

2.2   List of POSIX Calls

POSIX, which originated in the Institute of Electrical and Electronics Engineering (IEEE) standard committee for the Portable Application Standards Committee (PASC), refers to a Portable System Interface for compliant computer environments.

There are three parts to the implementation of the standard:

- • Definition of terms, global names and concepts,
- • Interface facilities, and
- • Data interchange format.

A list of POSIX function calls (APIs) appears in Appendix D. They are part of the interface and are required for use in the programming environment to effect portability which assures that programs can interface reliably with any POSIX-compliant operating system using the calls' specifications.

2.3   Standards References

The framework for major service areas of the TAFIM applies to the DII COE. Conversely, the DII COE is compliant with the TAFIM's classification of standard services.

The Computer-Aided Software Engineering (CASE) tools and environments specific to COE tools are compliant to the principles of the Portable Common Toolkit Environment (PCTE).

PCTE is the ISO/IEC 13719 Standard used with C and Ada Language bindings and also commonly referred to as ECMA 149 (PCTE).

Additional standards applicable to this document are related to the software development environment described in the Institute for Electrical & Electronic Engineers (IEEE) 1209 (Evaluation and Selection of CASE Tools), DoD-STD-1467 (Software Support Environment), NIST-ECMA 500-211 (Reference Model for SEE Frameworks), and MIL-HDBK-782 (Software Support Environment Acquisition).

With respect to software life cycle processes, the MIL-STD-498 *Software Development and Documentation*, and the ISO/IEC 12207 *Software Life Cycle Process* standards are applicable. For the Graphical User Interface (GUI) behavioral design, the MIL-STD-1801 *User Computer Interface* applies.

Standards for APIs are primarily based on Portable Operating Services Interfaces (IEEE POSIX P1003.n series). Additionally API standards created by various industry and consortia are adopted as "Public" APIs. These are X/Open (XPG4 Guide), Open Software Foundation (OSF/DCE), Network Management Forum (NMF) and other expert groups such XAPIA that feed the API standards process into the formal standards developing organizations.

Compliance to the above standards is recommended and the migration to these standards for life cycle development is an ongoing process in this and subsequent DII COE documentation.

3. DII COE Development Environment

3.1 Development Process Overview

This section describes the COE development process. The concept of automated integration is a powerful feature of the overall development process. This means automated tools are used to combine and load segments, make environmental modifications requested by segments, make newly loaded segments available to authorized users, and identify places where segments conflict with each other. Traditional system level integration then becomes primarily a task of loading and testing segments.

3.2 The COE Development Process

The COE development process consists of the following steps which are described in detail in the *DII COE Integration and Runtime Specification (I&RTS)* Chapter 3.0.

1. Submit Registration Form.
2. Develop Segment.
3. Download Segments for Submission.
4. Segment Integration Validation.
5. Segment Installation.

3.2.1 Segment Registration

Segment Registration is the entry point into the development process. Its purpose is to collect information about the segment for publication in a *segment catalog*. The DII segment catalog is available on-line through a Hypertext Markup Language (HTML) browser and contains information provided by developers in a segment registration form.

There are two steps that constitute the Segment Registration phase:

1. **Register the segment.** The segment registration form can be submitted in written form, through e-mail, or in HTML format. Appendix E of the *DII COE I&RTS* contains more information on how to do this.

   The segment registration form includes the following information:

   - Segment name,
   - Segment prefix,
   - Segment directory name,
   - List of related segments,
   - Program management point of contact,
   - Technical point of contact,
   - Process point of contact,

- Estimated memory required by the segment,
- Estimated disk storage requirements,
- Platform availability (PC only, Solaris only, etc.),
- Short paragraph describing the segment features,
- List of keywords for use in catalog searches, and
- Unclassified picture of the segment user interface (GIF, JPEG, or Xll Bitmap format).

Once the developer submits the registration form, the information is sent to the process point-of-contact. Segment information is entered into the segment catalog with a tentative release date for the segment. The segment prefix and directory requested will be granted unless they have already been assigned to another developer's segment.

2.   **Download segments required for development.** When notification is received that segment registration was successful, developers may download COE component segments, developer's toolkit, object code libraries, and other segments required for software development. The *DII COE Consolidated Version Description Document* and the *DII COE I&RTS* provide more information on how to download segments, tools, libraries, etc. The *DII COE I&RTS* also provides information on how to access and search the on-line segment catalog.

3.2.2   Segment Development

The COE approach is designed to be non-intrusive. Developers are free to establish a software development environment that is best suited for their project. The COE requires only that deliveries be packaged as segments, that segments are validated before submission, and that segments are tested in the COE prior to submission. The *DII COE I&RTS*, section 3.2.2 describes in detail the following steps of segment development. Also, refer to the *List of COE Tools and APIs* in Appendix B for DII COE Toolkit descriptions.

1.   Load and configure the COE.
2.   Verify that the COE is valid.
3.   Write and unit test code.
4.   Create segment structures.
5.   Validate the segment.
6.   Install and test the segment.
7.   Create a test installation tape (optional).
8.   Perform a system test.
9.   Test segment APIs.

3.2.3   Segment Submission

There are two steps in the automated process of submitting a segment. Reference the *Software Documentation Delivery Requirements, Version 1.0* for more information on Segment Specifications.

1. **Compress and encrypt the segment.** The tool *MkSubmitTar* performs this task on a "pre-MakeInstall" formatted segment. The directory *integ* must contain an annotated description of output from *VerifySeg*. When applicable, a test suite must be included for all APIs.

2. **Submit the segment**. The tool *submit* does this electronically across the Internet. Segments submitted via tape must be a relative *tar* of the output from *MkSubmitTar*, not the output of *MakeInstall*. Multiple segments can be delivered on the same tape provided that there is only one segment per physical *tar* tape segment.

3.2.4   Segment Integration

Segments received whether by tape or electronically are tested in isolation, and then tested as part of a deliverable system. Validation is performed at each step using the same tool set that the developer used during the development phase.

Prior to submitting a component to DISA, a developer must

- Package the component as a segment,
- Demonstrate COE compliance through tools and checklists,
- Test the segment in isolation with the COE,
- Provide required segment documentation, and
- Demonstrate the segment operating within the COE.

From this point on, the process steps are the responsibility of DISA, not the developer. The developer is still an active part of the process in isolating and correcting problems, therefore, the process steps are mentioned. Refer to the *DII COE I&RTS*, section 3.2.4 for a description of each of these process steps.

1. Receive segments.
2. Validate the segment.
3. Test segment in isolation.
4. Advance segment to test level.
5. Define variants.
6. Perform system test.
7. Accept segment.

3.2.5   Segment Installation

Segments can be distributed to sites either electronically or by other distribution media, as appropriate. The *MakeInstall* tool is used to extract segments and write them to tape or other media. The media is then manually delivered to the site. Installation can also be performed electronically through the *RemoteInstall* tool. The *RemoteInstall* tool operates in either a "push" or "pull" mode. In a push mode, DISA initiates electronic transfer of segments to operational sites. In a pull mode, the remote site initiates the segment transfer. Once received at a site, the site administrator can use the installation tools in the System Administration application to load segments directly onto individual workstations or to multiple workstations as "segment servers."

3.3   Migration Considerations

Much of the present and planned functionality of the DII COE is derived from existing legacy systems, not new development, as it is often not feasible to totally abandon a system to start over. Legacy systems must implement a migration strategy that allows them to take advantage of COE benefits. The strategy must simultaneously balance: full COE compliance versus implementation cost; rapid system deployment versus risk to system stability; porting functionality versus new development; and presentation of capabilities users already have versus duplication.

It is helpful to remember that the development approach is to build on top of the DII COE, not to decompose the COE into constituent parts to then build on top of other architectures or body of software. In other words, the approach is to integrate components from legacy systems into the COE, not to integrate the COE into an existing legacy system. This perspective is fundamental to successful integration.

The key to reusing the COE, and to achieving COE compliance, is the concept of "Public" APIs. APIs represent the linkage through which segments may gain access to COE services. Software developers and integrators must build to public APIs rather than to proprietary versions of software products, since the public APIs will be preserved as the COE evolves. Applications must migrate away from private or legacy APIs since they will not be supported in COE releases.

The following considerations will lead to a successful migration strategy when integrating components from a legacy system into the COE. This information, as well as additional details, may be found in the *DII COE I&RTS*, section 3.3.

- Create a requirements matrix.
- Develop a schedule for achieving Level 8 compliance (Full COE Compliance Level).
- Determine how the segment will be integrated with the Executive Manager.
- Determine which account group(s) the segment will belong to.
- Determine the required runtime environment extensions.
- Identify support services within the legacy system.
- Identify public DII COE APIs to be used.
- Negotiate new APIs or modifications with the DISA Engineering Office.

- Build only to public COE approved APIs.

3.4 Development Environment

The COE development environment allows a variety of the development tools and processes so long as the end products comply with the COE and integrate with the overall system. Even the developers' own internal tools and processes are allowed if the end products do not violate the COE architecture and do not inhibit automated integration, interoperability, and portability.

The technical requirements and specifications of the COE development environment are fully described in the *DII COE I&RTS*, Section 6, Development Environment. The following paragraphs provide a brief overview of this information.

3.4.1 Development Scripts

The development scripts are the scripts that developers use in the COE development environment. Since the COE distinguishes the development scripts from the runtime scripts, procedures, similar to those suggested in the *DII COE I&RTS*, must be established to ensure development scripts are stripped from the final delivery.

3.4.2 Directory Structures

The COE allows developers to create any logical directory structures for use during the development process. However, COE installation tools enforce the COE directory structure as specified in section 5 of the *DII COE I&RTS*. Logical directory structures, like the one described in section 6 of the *DII COE I&RTS*, are the structures that provide benefits not only in software development but also in software maintenance and distribution.

3.4.3 Private and Public Files

The private files are the files that are accessible only to the COE segment that owns the private files, whereas the public files are open to any COE segments. Developers are expected to hide the process details and data elements of their segments by creating private files and placing high-level functions in the public files. This method minimizes the interdependencies of COE segments, which in turn minimizes the scope of code changes when a segment is changed. This design principle is based on object-oriented programming concepts, e.g. data abstraction and data hiding.

3.4.4 Developer's Toolkit

The COE provides developers with a toolkit to assist in developing COE compliant applications. The toolkit includes API libraries, C header files, on-line help files, and royalty- and license-free COE development tools. Since the toolkit is not part of the final COE-based system, it is not in the standard COE segment format required by the COE installer.

3.5 COE Tools Overview

The COE tools were developed to perform selected DII COE tasks. The tools may be executed from the command line, menus, or a code segment using published APIs. Detail information on the use of the COE tools and how they communicate with the outside world are available from the *DII COE I&RTS, Appendix C*. A list of the COE tools available that help developers create, install, and test segments is provided in Appendix B of this document. In all cases, the final authority on the use of a tool is the associated API reference guide.

3.6   API References

The DII COE, by its very nature and design, is transitional and evolutionary. Information available concerning the DII COE APIs can be found in the DII COE documentation listed in Appendix B of the *DII COE Programmer's Reference Manual*. For detailed information on APIs, refer to the *JMTK Developers Manual*, the *DII COE Application Programmer Interface (API) Reference Guides*, the *DII COE User Profiles API*, the *DII COE Programming Guides*, the appropriate Version Description and API Documents.

4. UNIX-Based Components Description

The COE is functionally comprised of: the Kernel, Developer's Toolkit, Infrastructure Services, and Common Support Applications.

4.1   DII COE Kernel (UNIX)

The DII COE Kernel (Version 3.0.0.2) for UNIX provides the core services for the SUN SPARC workstation and the HP Tactical Advanced Computer (TAC) types 3 and 4. It is the minimum set of software required for the operation of any DII COE workstation regardless of how the workstation is to be used. The Kernel is comprised of: the OS, Windowing, System Administration, Security, Print Manager, Runtime Tools, and Distributed Computing Services.

4.1.1   Operating Systems

The operating system (OS) delivered with the COE consists of the basic, vendor supplied operating system software and any OS patches required to operate the COE. For SUN SPARC platforms with SPARC STORAGE Arrays (SASs), Volume Manager 2.1 is included as part of the Solaris operating system. The supported UNIX-based OS are Solaris 2.4, Solaris 2.5.1 and HP-UX 9.

4.1.1.1   Solaris 2.4 and Solaris 2.5.1

Platform: SUN SPARC

Description: POSIX-compliant operating system which provides basic infrastructure services required to operate a SUN SPARC workstation.

Purpose: The operating system allocates system resources between various processes, manages physical storage, provides a basic user authentication service and processes external devices attached to the workstation.

Usage: Most operating system commands are executed directly from the command line. However, there is a common set of POSIX calls which are listed in appendix D. These calls enable the programmer to:

- Create, execute, and terminate processes; communicate between processes; and coordinate process operations,
- Create, delete and process files and directories,
- Activate a process on a timed basis or in response to an external event,
- Manage configurable system parameters, and

- Access basic security functions inherent in the operating system.

References: Vendor documentation

4.1.1.2   HP-UX

Platform: HP 9.0.7

Description: POSIX-compliant operating system which provides basic infrastructure services required to operate an HP TAC 3 (models HP 715, 720, 730, 735, 750, 755) or TAC 4 (models HP 712, 770) workstation.

Purpose: The operating system allocates system resources between various processes, manages physical storage, provides a basic user authentication service and processes external devices attached to the workstation.

Usage: Most operating system commands are executed directly from the command line. However, there is a common set of POSIX calls which are listed in Appendix D. These calls enable the programmer to:

- Create, execute, and terminate processes; communicate between processes; and coordinate process operations,
- Create, delete and process files and directories,
- Activate a process on a timed basis or in response to an external event,
- Manage configurable system parameters, and
- Access basic security functions inherent in the operating system.

References: Vendor documentation

4.1.2   Windowing

The DII COE windowing for UNIX-based systems include three layered components:

- X-Window (X11R5)
- OSF/MOTIF
- Common Desktop Environment (CDE)

4.1.2.1   MIT X-Window (X11R5)

Name: X-Window (X11R5)

Platform: Solaris 2.4 and 2.5.1
                   HP-UX 9.07

Description: Protocol describing the interface between an application and the application's presentation logic to high-performance device-independent graphics, and a hierarchy of resizable, overlapping windows.

Purpose: The X-Window system enables graphical user interfaces in a networked environment using a client/server architecture. The server resides on the workstation and produces the graphics and collects keyboard and pointing device data. The client program performs the computations and resides anywhere on the network.

Usage: X-Window provides an intrinsic toolkit "XM" which provides the mechanism for windowing functions.   Some aspects of an X-Window session are configurable. Various initialization files allow:

- Keyboard mapping - associate workstation keystrokes with client keystrokes,
- Fonts - select the look for characters displayed on the screen, and
- Colors - allows selection of the basic color scheme of the window. Management of colors is also accomplished using the Desktop's Style Management application.

References: Vendor documentation, and other sources, including frequently asked questions (FAQs) are available on the Internet:

- http://www.uwsg.indiana.edu/usail/library/x.html,
- http://www.x.org (X consortium), and
- http://www.osf.org/og (Open Group).

4.1.2.2   MOTIF

| **Name:** | MOTIF 1.2.2 |

| **Platform:** | Solaris 2.4 and 2.5.1 |
| | HP-UX 9.07 |

Description: An X-Window, window manager client application developed by the Open Systems Foundation (OSF) providing a graphical user interface.

Purpose: Motif includes the Motif Toolkit (also called "XM" or the "Motif widgets"), which enforce a policy on top of the X Toolkit Intrinsics ("XT"). This policy determines the specific "look and feel" of the windowing environment. Motif enables an icon and menu-driven interface between applications and the workstation user/operator.

Usage: Motif is made up of four parts: a User-Interface Guideline, an API toolkit of "C" routines which helps in the building of applications which conform to the Guideline, the Motif Window Manager (MWM), and a language UIL which is designed to ease user interface development.

References: Vendor documentation, and other sources, including Frequently Asked Questions (FAQs) are available on the Internet.

- http://www.osf.org/motif (Open Systems Foundation), and
- http://www.cen.com/mw3 (Century Computing).

4.1.2.3   Triteal CDE 4.0

| **Name:** | TED 4.0 |

| **Platform:** | Solaris 2.4 and 2.5.1 |
| | HP-UX 9.07 |

Description: The desktop for the COE is based on specifications jointly developed by a group of vendors. The TriTeal Enterprise Desktop (TED) is provided as part of the Kernel on UNIX systems.

Purpose: The TriTeal Enterprise Desktop provides a uniform basic desktop for all UNIX-based platforms. The desktop provides modules to implement basic functionality:

- Work space management,
- Text editing,
- Simple mail,
- Calendar and schedule management ,
- Terminal emulation,
- Calculator function,
- File management,
- Printer management,
- Style management,
- Session management
- Login management,
- Process communications, and
- Application integration.

Usage: The desktop environment can be customized to include application segments. Both icons and menu items can be added. Programmers may embed a menu description entry in the `SegInfo` file for the segment to define new pull down menus or extensions to existing menus. Programmers may also add icons to the desktop using a similar method. In order to maintain a common look and feel to desktops and applications across the COE, a style guide has been established which should be followed when implementing a segment or application.

References: Vendor documentation
DII COE Programming Guide (2.0.0.1)
User Interface Specifications for the DII (2.0)

4.1.3   System Administration

The DII COE System Administration (SA) consists of various tools. Detailed information about these tools will be provided in subsequent releases. The SA consists of the following components:

- Session Manager, and
- Process Manager.

The System Administrator is a menu driven application that performs common system management functions performed by a system administrator. The application also contains the security management menu interface described in section 4.1.4.5.

The menu interface allows the system administrator to:

- Select printers, configure printers, and manage print jobs,

- Shutdown or restart the system, mount file systems, and initialize hard or floppy disks,
- Load or install COE compliant segments, and
- Configure networking parameters on the system.

4.1.3.1   Session Manager 1.0

Name: Session Manager 1.0

Platform: Solaris 2.4 and 2.5.1
                     HP-UX 9.07

Description: TBD

Purpose: TBD

Usage: TBD

References: TBD

4.1.3.2   Process Manager 1.0

Name: Process Manager 1.0

Platform: Solaris 2.4 and 2.5.1
                     HP-UX 9.07

Description: TBD

Purpose: TBD

Usage: TBD

References: TBD

4.1.4   Security

The DII COE Security Services components are used in the kernel. It provides security features and mechanisms both for the client and the server. The Security Services are:

- System Services, and
- Security Manager.

4.1.4.1 Client Server Environment - System Services (CSE-SS)

Name: CSE-SS 1.2

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: The System Services segment of the DODIIS Client Server Environment provides basic utility services to enhance security.

Purpose: Four CSE-SS utilities are implemented as security enhancements in the COE Kernel:

- Console - Opens a window for the display of information, no input is allowed,
- Deadman - Locks a terminal after a configurable period of time to preclude unauthorized access to an unattended terminal,
- Password - Allows the user to change logon password within the desktop environment, and
- X-Window Display Manager - Manages a collection of X displays, authenticating users and running sessions.

Usage: Primarily designed to enhance existing security features of the UNIX operating system.

References: TBD

4.1.4.2 Security Manager 1.0

Name: Security Manager 1.0

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: The Security Manager is primarily a menu driven application which allows the system security staff to manage users and profiles in the system. The application calls or APIs for Security Manager have been implemented to parallel the command line interface provided by the Security Manager. These APIs are referred to as the user profile APIs.

Purpose: The user profile APIs allow the programmer to create an application that:

- Controls access to applications through the use of roles,
- Controls access of individuals to the system, and
- Manages parameters related to a user session.

In order to use the APIs which control access to the system, either the user or the application must be authorized.

Usage: User profile APIs will most commonly be called from within an application to:

- Retrieve the UNIX identification number assigned at logon,
- Extract information from the User Profile database,
- Save current profile information for use in subsequent logons,
- Provide an explanatory text string associated with an error message, and
- Replace non-alphanumeric characters with underscores ('_').

Command line functions can be used to display the active role for a user and to switch between roles where the user has active permission grants.

References: DII COE User Profiles Application Programmer Interface (3.0.0.2)

4.1.5   Print Manager 1.0

Name: Print Manager 1.0

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: The Print Manager provides basic applications programmer interfaces (APIs) which provide the means of controlling printers and printing data from within an application.

Purpose: The Print Manager APIs allow the programmer to create an application that:

- Requests a printer, specifies print context, and initiates printing,
- Transfers text strings and files to a printer, and
- Determines the printing environment.

Usage: Print Manager APIs are called from within an application when it is required to:

- Initiate connection to a printer,
- Write a string or array of strings to the connected printer,
- Terminate a print job and terminate connection to a printer,
- Send a file or text buffer to a printer selected by the operator, and
- Return information about a printer.

Command line versions to return printer information are also provided.

References: DII COE Application Programmer Interface (API) Reference Guide Version 3.0.0.3

4.1.6   COE Runtime Tools 1.0

Name: Runtime Tools 1.0

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: The runtime tools developed for the COE provide the basic method used to install segments in the DII COE environment.

Purpose: The runtime tools define the toolset used to install segments. They provide special functionality to script execution in UNIX to support pre- and post-install activities. Information regarding installed segments can be displayed as required. Additionally, the system administrator can determine the association between segments and authorized users and user groups.

Usage: Although primarily designed to support the system administrator, the applications developer must use these basic tools in constructing any scripts run prior to or after the actual installation. COE Runtime Tools are normally executed on the command line or as a menu item in the system administrator application. Some of the COE Runtime tools do provide functionality through APIs which are of general use. They allow an application to:

- Solicit a yes/no type response from the user,
- Display a window with a general message or a termination message,
- Prompt the user for a text response, and
- Prompt for passwords with a verify option.

References: *DII COE I&RT*-Appendix C.

DII COE Application Programmer Interface (API) Reference Guide
(HP and Solaris Version 3.0.0.3)

4.1.7   Distributed Computing Services - Transarc DCE Client 1.1

Name: Transarc DCE 1.1 (Client)

Platform: Solaris 2.4 and 2.5.1
                     HP-UX 9.07

Description: The DOE is an open systems solution to distributed computing. The
                     DCE is composed of a series of services including Core services
                     which are provided as part of the COE Kernel. Additional services,
                     which vary across platforms, are also available. These Extended
                     DCE services are described in section 4.3.6 (DCE Server, DCE
                     Distributed File System (DFS), and Cell Manager).

Purpose: DCE provides a communications environment to hide the
                     complexity of network communications from the application
                     developer. This is accomplished by placing layers of services on top
                     of the network services. The DCE Core services support:

                     • Programs performing many actions concurrently (Threads),
                     • Direct calls to procedures on remote systems (Remote
                       Procedure Calls or RPC),
                     • Logically naming objects within a DCE cell (Cell Directory
                       Service or CDS) or throughout a network (Global Directory
                       Service or GDS), and
                     • Authentication of users and determination of privileges
                       across machines.

Usage: DCE simplifies the process of developing distributed applications.
                     DCE Threads are based on the POSIX threading standard and do
                     not introduce any new complexity into the process. RPC calls
                     eliminate explicit programming of network communications and
                     mask differences in data representation across platforms. The
                     directory services allow resources to be identified by name only,
                     without reference to physical location.

References: Vendor documentation, and other sources, are available on the
                     Internet:

                     • http://www.transarc.com (Transarc's homepage) , and
                     • http://www.osf.org/dce (Open Systems Foundation).

4.2   DII COE Developer's Toolkit

The DII COE developer's toolkit helps programmers with the creation, installation, testing and management of segments.

4.2.1   COE Developer's Toolkit

Name: Developer's Toolkit

Platform: Solaris 2.4, 2.5.1
                        HP-UX 9.07

Description: These are DII COE tools that the programmer can use to create, install and test segments.

Purpose: To aid in the development of segments and to standardize procedures for loading segments.

Usage: The Developer's Toolkit will be used by programmers and software integrators to:

- Compute space required by a segment,
- Test a segment,
- Covert a segment to the latest format,
- Create a descriptor file,
- Write segments to an installation medium or package segments,
- Temporarily install a segment,
- Remove a segment,
- Time stamp a descriptor,
- Validate a segment, and
- Update segment data in the descriptor.

References: *DII COE Programmer's Manual,*
                        DII COE I&RTS,
                        DII COE Programmer's Guide, and
                        Version Description Document DII COE Developer's Toolkit (HP and Solaris)

4.3   COE Infrastructure Services

The DII COE Infrastructure Services tools consist of Systems Management and Network Management.

4.3.1   Systems Management

The COE Systems Management Infrastructure Services provide optional services that simplify management of a client/server network of heterogenous workstations and other desktop systems. It is a general purpose utility to assist the system manager in configuring and maintaining UNIX-based workstations.

4.3.1.1   Tivoli Management Environment 3.0

Name: Tivoli Management Platform 3.0.1

Platform: Solaris 2.4

Description: General purpose toolset to assist system administrators in managing a client/server computing environment.

Purpose: Tivoli Management provides enhanced system administrator functionality over the Kernel and provides a base for additional Tivoli utilities. The Management Environment allows the system administrator to:

- Explicitly identify system administrators and the system administration resources the administrator is allowed to access, as well as the privileges allowed when accessing those resources. Administration functions are performed without requiring the root password.
- Log administrator activity into application-specific notification groups, providing an administration-specific audit trail.
- Manage common system resources (e.g., user accounts, host name space, etc.) according to a resource template, or profile.
- Perform management actions on remote client nodes, including querying its physical and logical configuration without having to log into that node.
- Establish default policy for a new instance of a resource and validation policy to check that existing resources still comply with the rules.
- Interconnect managed resources to provide scalable system administration and to reduce the effects of individual management server failure.
- Define administrative procedures for one or more platforms, permitting common system administration tasks, such as removing core files, pruning out unused user accounts, or

checking for invalid user passwords, to be performed securely.

- Perform certain actions at regular intervals or at off-hours so that they do not interfere with mission-critical operations.

Usage: The Management Environment is used by the system administrator in configuring, operating, and enhancing a more robust client/server computing environment.

References: Vendor Documentation. Limited information can also be found at Tivoli's homepage, http://www.tivoli.com.

4.3.1.2   Tivoli Administration 3.0

Name: Tivoli Admin 3.0

Platform: Solaris 2.4

Description: System Administration utility providing user, group, and host name space management. Provides Network Information Service (NIS) domain management.

Purpose: Allows the systems administrator to:

- Add, edit, and delete user accounts, as well as distributing user account information to other profile managers, to managed nodes, and to NIS and Extended NIS (NIS+) domains;
- Add, edit, and delete UNIX groups, as well as distributing group information to other profile managers, to managed nodes, and to NIS and NIS+ domains;
- Add, edit, and delete host name-IP assignments, as well as distributing host name space information to other profile managers, to managed nodes, and to NIS domains;
- Add, edit, and delete NIS maps and map data, as well as making and pushing NIS maps; and
- Distributes user and group profiles in an NIS+ environment.

Usage: Tivoli Admin is used by the system administrator in configuring, operating, and enhancing a more robust client/server computing environment.

References: Vendor Documentation. Limited information can also be found at Tivoli's homepage, http://www.tivoli.com

4.3.1.3   Tivoli Courier 3.0

Name: Tivoli Courier 3.0

Platform: Solaris 2.4

Description: System Administration utility to assist the system manager in configuring and maintaining a workstation over the network.

Purpose: Tivoli Courier provides software distribution and removal across a client/server network of heterogeneous platforms.

Usage: Tivoli Courier provides for centralized control of a workstation software configuration.

References: Vendor Documentation. Limited information can also be found at Tivoli's homepage, http://www.tivoli.com.

4.3.1.4   Tivoli Sentry 3.0

Name: Tivoli Sentry 3.0

Platform: Solaris 2.4

Description: System Administration utility to monitor the client/server environment.

Purpose: Tivoli Sentry runs on local nodes to monitor system statistics and to detect specified conditions.

Usage: Tivoli Sentry provides an overview of the status of a computing environment. When certain events are detected, Sentry can be configured to initiate notification actions or to take designated corrective actions.

References: Vendor Documentation. Limited information can also be found at Tivoli's homepage, http://www.tivoli.com.

4.3.2   Network Management

Infrastructure network management services provide additional capabilities for managing server and network activity from a client computer. Standards for Network Management (Management Information Bases or MIBs) are community developed as outlined in section 5 of the *Programmers Manual*.

4.3.2.1   Empire 1.0.0.1

Name:	Empire UNIX Systems Management Agent 1.35.03

Platform: Solaris 2.4
HP-UX 9.07

Description: Network and System Administration utility to manage the client/server environment.

Purpose: Empire implements a network management MIB and a host resource MIB. It also implements a UNIX specific extension to the host resource MIB. Empire provides statistics on:

- Network configuration and statistics,
- Host configuration and status,
- File system configuration,
- Currently running software and performance,
- Installed software on certain platforms,
- Kernel and boot configuration, and
- UNIX operational and internal statistics.

Usage: Empire is used by network and system administrators to operate a client/server environment and monitor system performance.

References: Vendor Documentation. Limited information can also be found at Empire Technologies homepage, http://www.com/empiretech/acapabilities.html/#unixmib.

4.3.2.2   NetMetrix 1.0.0.1

Name: Hewlett Packard NetMetrix 4.5.0

Platform: Solaris 2.4
HP-UX 9.07

Description: Network Administration utility to remotely manage the client/server environment.

Purpose: NetMetrix implements a remote network management MIB. NetMatrix provides statistics on:

- Network faults and problems,
- Network performance and throughput for capacity planning,
- Network accesses, and

- Network availability.

Usage:	NetMetrix is used by network administrators to remotely monitor a client/server environment and monitor performance.
References:	Vendor Documentation. Limited information can also be found at HP's homepage, http://www.tmo.hp.com/tmo/ntd

4.3.2.3   Streams 1.0.0.0

Name:	Hewlett Packard Streams 1.0.0.0
Platform:	HP-UX 9.07
Description:	General purpose communications utility designed to systematize existing UNIX character I/O mechanisms and support the development of communications services.
Purpose:	Streams provides a general, flexible facility and set of tools used for developing UNIX communications services.
Usage:	Streams provides low-level functionality for communications programming.
References:	Vendor Documentation.

4.3.3   Print Management

Name:	Sun NewsPrint 2.5
Platform:	Solaris 2.4
Description:	General utility providing enhanced functionality for network printers.
Purpose:	NewsPrint provides network printing support and utilities as well as print file filters and fonts.
Usage:	NewsPrint is used to configure a SUN SPARC printer.
References:	Vendor documentation.

4.3.4   Communications

Communications infrastructure services provide the capability to interact with other DoD message and data systems. Communications services provide basic communications support to various Common Support Applications (Section 4.4).

4.3.4.1   Unified Build Communications Service - UB 3.0.2.2

Name: Unified Build (UB) Comm Service 3.0.2.2

Platform: Solaris 2.4
                       HP-UX 9.07

Description: Utilities providing the common communications interface to tactical and message traffic systems.

Purpose: UB Comm Services accepts input from Link 11, Link 14, navigational (Global Positioning System type) interfaces, and other special interfaces and decodes them for use in applications. Using the UB Comm Services, data can be obtained from Electronic Intelligence (ELINT) sources, Tactical Digital Information Links (TADIL), and from Joint Surveillance Target Attack Radar System (JSTARS).

Usage: The UB Comm Service is used primarily by other applications in the Unified Build, but may be used in other applications where appropriate interfaces are used.

References: Comms Service Application Programmers Interface (API) for the DII COE Software Development Environment (SDE).

4.3.4.2   USA Comm Server 1.0

Name: Communications Software 1.4.

Platform: Solaris 2.4

Description: Utilities providing the common communications infrastructure for the Army Common Hardware and Software program.

Purpose: Communications Software (CS) provides reliable message delivery through Tactical Communications Interface Modules (TCIMs), Tactical Radio, LANs and WANs. The Communications Server supports both DCE and non-DCE network environments. The CS

enables clients without TCIMs to send/receive messages through TCIMs attached to other servers.

Usage: Communications Software is used to interface applications with DoD tactical communications networks.

References: COE Communications Server Abstract

4.3.5   Data Management

Data management are COTS products that provide data management services to DII COE applications. The data management services include the high-speed access to and manipulation of information such as: character strings, numbers, money, and date/times. In addition, the data management system facilitates the exchange of data between inter-connected computers.

4.3.5.1   Informix

Name: INFORMIX-OnLine Dynamic Server, 7.2 for UNIX

Platform: Solaris 2.4

Description: Informix is a multi-threaded database server designed to exploit the capabilities of both Symmetric Multi-Processor (SMP) and uniprocessor architectures to deliver high database scalability, manageability, and performance.

Purpose: Informix is a COTS database server that provides data management services to applications.

Usage: Informix provides the means for application programs to store and retrieve information. It also provides operational support systems and database administrator functions.

References: Vendor's product manuals and third-party technical books.

4.3.5.2   Oracle

Name: Oracle Server, 7.2.2.4

Platform: Solaris 2.4
HP-UX 9.07

Description: Oracle is a multi-threaded, multi-server database server that delivers high scalability, availability, and performance needed for mission-critical online transaction processing systems.

Purpose:	Oracle is a COTS database server that provides data management services to applications.
Usage:	Oracle provides the means for application programs to store and retrieve information. It also provides operational support systems and database administrator functions.
References:	Vendor's product manuals and third-party technical books.

4.3.5.3   Sybase

Name:	Sybase SQL Server, 11
Platform:	Solaris 2.4 HP-UX 9.07
Description:	Sybase is a fully symmetrical and multi-threaded database server that delivers high scalability, availability, and performance needed for mission-critical Online Transaction Processing (OLTP) or decision-support systems.
Purpose:	Sybase is a COTS database server that provides relational data management services to applications.
Usage:	Sybase provides the means for application programs to store and retrieve information. It also provides operational support systems and database administrator functions.
References:	Vendor's product manuals and third-party technical books.

4.3.6   Distributed Computing Services

DCE infrastructure services complete the basic functionality provided in the kernel by supporting a DCE server and provide extended file and cell management services.

4.3.6.1   DCE Server

Name:	DCE Server 1.1
Platform:	Solaris 2.4 HP-UX 9.07
Description:	DCE Server provides a central point for configuring and managing a DCE cell.

Purpose: DCE server must be installed and active on a system prior to adding any clients to the DCE cell. Cell configuration data is maintained on the servers. DCE Servers also run the Enhanced File Services to support DCE DFS.

Usage: DCE Server is used if DCE is to be provided and a cell activated.

References: Vendor documentation. Transarc maintains a web site at http://www.transarc.com.

4.3.6.2   DCE Distributed File System (DFS)

Name: Distributed File System (DFS) 1.1

Platform: Solaris 2.4

Description: DFS enables collections of workstations to share files as a single unit. DFS is built on top of the core DCE services provided in the Kernel.

Purpose: DFS provides location-independent access to data files. It can be configured to provide extra security by replicating critical files across several server machines to minimize the impact of hardware failures.

Usage: DFS is used to manage and share data within a site or throughout the COE.

References: Vendor documentation. Transarc maintains a web site at http://www.transarc.com.

4.3.6.3   DCE Cell Manager

Name: Cell Manager 1.1

Platform: Solaris 2.4

Description: Cell Manager is a centralized GUI tool for managing a DCE cell.

Purpose: Cell Manager allows network, system, and security administrators to configure a DCE cell. It provides the following tools:

- **Namespace Manager** - Views, searches and edits the cell directory namespace;

- **Security Manager** - Views, searches, and maintains the security registry; and
- **Configuration Manager** - Monitors and manages host status.

Usage: Cell Manager is used to simplify management of DCE cells.

References: Vendor documentation. Chisholm Tech maintains a Cell Manager web site at http://www.chistech.com/products/dce-cellmgr.

4.3.6.4   Netscape Browser 2.0.0.2

Name: Netscape Navigator 2.0.1

Platform: Solaris 2.4
HP-UX 9.07

Description: Netscape Navigator is a COTS graphical client application for accessing the World Wide Web (WWW) and other Hypertext Transport Protocol (HTTP) networks.

Purpose: Netscape Navigator provides graphical tools for accessing the WWW and searching, reviewing, and retrieving information from available sources, which include government sites, commercial sites, and educational sites, to name a few. It also supports electronic mail and access to newsgroups.

Usage: Netscape Navigator is used to access WWW sites on the SIPRNet using Netscape Navigator capabilities. This allows access to the SIPRNet Web, SIPRNet Newsgroups, and SIPRNet e-mail. Installation and usage require Netscape Navigator 2.0.1 licenses.

References: Netscape Communications Corp.
Netscape on-line documentation (http:/www.netscape.com)
DII System Administrator's Manual

4.3.6.5   Netscape NewsServer 1.0.0.2

Name: Netscape NewsServer 1.12

Platform: Solaris 2.4
HP-UX 9.07

Description: Netscape NewsServer is a COTS software package for creating a news search server.

Purpose:	Netscape NewsServer provides support for hosting news groups on the Internet or Internet-compatible networks.
Usage:	Netscape NewsServer is used to support group discussions in the COE environment or to use for search applications.
References:	Netscape Communications Corp. Netscape on-line documentation (http:/www.netscape.com) *DII System Administrator's Manual*

4.3.6.6   Netscape WebServer 1.0.0.1

Name:	Netsite Web Server 1.12
Platform:	Solaris 2.4 HP-UX 9.07
Description:	Netsite Web Server (WEBSv) is an HTTP World Wide Web server application from Netscape Communications Corp. It is a COTS segment.
Purpose:	Netsite Web Server provides support for hosting HTTP WWW sites and HTTP intranet sites.
Usage:	Netsite Web Server accepts and fulfills requests for HTML pages and other data hosted by DII COE HTTP network sites. Installation and usage of WEBSv 1.12 require Netsite Web Server v 1.1 licenses.
References:	Netscape Communications Corp. Netscape on-line documentation (http:/www.netscape.com) *DII System Administrator's Manual* *Netsite Web Server Installation Procedures - Quick Guide* (delivered with the segment)

4.3.6.7   News Make Group 1.0.0.1

Name:	News Make Group 1.0.0.1
Platform:	Solaris 2.4 HP-UX 9.07
Description:	COTS product. Information not available at this time.
Purpose:	TBD

Usage: TBD

References: Vendor documentation. TBD

4.3.7   General Utilities

General utilities provide generic capabilities to COE platforms.

4.3.7.1   FTP

Name:	File Transfer Protocol (FTP) Tool
Platform:	Solaris 2.4 and 2.5.1
Description:	The FTP tool provides a graphical user interface to allow file transfers between systems.
Purpose:	Using the FTP tool, information (files) can be pulled from or pushed to external computer systems.
Usage:	FTP is used to transfer files that are not accessible using DCE services.
References:	Online Help available on the Internet.

4.3.7.2   GZIP

Name:	GZIP 1.2.4
Platform:	Solaris 2.4 and 2.5.1 HP-UX 9.07
Description:	The GZIP software compression/decompression utility is similar to the PKZIP utility used on MS-DOS based computers.
Purpose:	GZIP allows for the reduction in size of a file, thereby reducing the amount of disk storage or reducing transmission times for files sent over the network.
Usage:	GZIP can be used to compress files. GUNZIP can expand files compressed by GZIP in addition to files compressed using the LZW compression algorithm.
References:	Online documentation. The Free Software Foundation which developed GZIP is located on the Internet at http://www.gnu.org.

4.3.7.3   Practical Extraction and Report Language (Perl)

Name: Perl 5.0.0.2

Platform: Solaris 2.4 and 2.5.1
                 HP-UX 9.07

Description: Perl is a C-like Interpretive language optimized for scanning text
                 files, extracting information, and printing reports.

Purpose: PERL extends the capabilities provided by the sed, awk, or sh
                 UNIX utilities. Perl does not have limitations on the size of data
                 other than the amount of memory available.

Usage: Perl is primarily used for systems management tasks.

References: Numerous sources are available on the Internet:

- http://perl.com/perl/index.html (The Perl Homepage), and
- http://www.www.metronet.com/0/perlinfo/perl5/manual/per
  l.html (Perl documentation).

4.3.7.4   Tool Command Language (TCL)

Name: TCL 7.5 / Tk 4.1

Platform: Solaris 2.4 and 2.5.1
                 HP-UX 9.07

Description: TCL is a high level scripting language running in a UNIX
                 environment.

Purpose: TCL is a basic programming language with features of conventional
                 languages. Tk is a toolkit of widgets which provides GUI
                 capabilities to TCL.

Usage: TCL is suitable for small, simple programs which require a good
                 user interface.

References: Numerous sources are available on the Internet. An index to
                 Internet TCL resources can be found at:
                 http://www.sco.com/Technology/tcl/TCL.html.

4.3.7.5   WABI 2.0 and 2.2

Name:	Windows Application Binary Interface (WABI)
Platform:	Solaris 2.4 (Version 2.0 only) (Version 2.2 is shipped with Solaris 2.5.1)
Description:	WABI is a UNIX application that enables Microsoft Windows® applications to run on the Solaris desktop.
Purpose:	WABI translates Windows® calls to Dynamic Link Libraries (DLL) into Solaris and X-Window calls. This enables Windows® applications to execute and integrate into the Solaris desktop environment. WABI supports networking via WinSock and textual clipboard operations between the two environments. The applications are also integrated into the CDE.
Usage:	WABI is used solely to allow a Windows® application to run on a Sun workstation. Sun maintains a list of applications that are certified to run under WABI. Other lists of applications that "have been found to work properly" are also available. These lists are specific to WABI versions. Certified operation in version 2.2 may not indicate operation in version 2.0.
References:	Numerous sources are available on the Internet, however, most sites are transitioning to later versions of WABI. A repository of information about applications which run under WABI is found at http://wabiapps.psgroup.com.

4.3.8   Security Utilities

Security utilities provides a set of tools to assist security administrators in correcting vulnerabilities and preventing unauthorized access.

4.3.8.1   Security Administrator Tool for Analyzing Networks (Satan) 1.0.01

Name:	Satan 1.1.1 (Freeware)
Platform:	Solaris 2.4 and 2.5.1 HP-UX 9.07
Description:	Satan attempts to determine if known security vulnerabilities are present on a system.

Purpose: Satan attempts to exploit known security problems across the network. Satan remotely probes the Transmission Control Protocol (TCP) ports of a system and makes an educated guess about the service a port supports. It then attempts to gain access using known vulnerabilities of the service.

Usage: Satan is run by a security administrator from a remote machine to detect security vulnerabilities.

References: Online documentation.

4.3.8.2 Courtney 1.0.0.1

Name: Courtney 1.0

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: Courtney detects if a Satan-like probe has been run against a system.

Purpose: Courtney indicates that an individual (probably unfriendly) has run Satan against the host system. It monitors the computer network for the connection attempt pattern used by Satan.

Usage: Courtney is used by security managers to report attempts to break into a system.

References: Online Documentation. An additional information and reference manual available at the Computer Security Technology Center, http://ciac.llnl.gov/cstc/CSTCIntroduction.html.

4.3.8.3 Crack 1.0.0.1

Name: Crack 4.1

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: Crack attempts to break each entry in the UNIX password file.

Purpose: When executed, Crack uses a set of dictionaries and applies a set of rules to detect passwords that may lead to a security risk. Candidate passwords are encrypted and then compared to system password files. The dictionaries and the rules can be customized.

Usage:	Crack is used by security administrators to detect passwords that are prone to being guessed. Running Crack requires a large amount of system resources.
References:	Online Documentation

4.3.8.4   Security Profile Inspector (SPI) 1.0.0.1

Name:	SPI 3.2.2
Platform:	Solaris 2.4 and 2.5.1
Description:	SPI is a suite of automated security inspection utilities.
Purpose:	SPI provides security inspection tools that attempt to detect security vulnerabilities:

- **Access Control** - Attempts to achieve an input goal by exploiting weaknesses in file permissions, dubious file ownership, and user group memberships.
- **Binary Authentication** - Attempts to verify system object authenticity and patch currency based on supplied authentication tables.
- **Change Detection** - Detects changes in files based on changes in file attributes; identifies new or missing files; detects similar changes in user and user group accounts.
- **Password Security** - Attempts to penetrate user accounts by generating variants of words from the user's account and from customized dictionaries. Also, tests for password aging and bypasses retesting of unchanged passwords.
- **System Profile** - Conducts specialized tests for operating system vulnerabilities including weaknesses in network services, system configuration files, and root auto-exec files.
- **Configuration Queries** - Allows flexible inquiries into objects of computer system security.

Usage:	SPI is used by security and system administrators to help keep systems secure.
References:	An additional information and reference manual available at the Computer Security Technology Center, http://ciac.llnl.gov/cstc/CSTCIntroduction.html.

4.3.8.5 TCP Wrappers 1.0.0.1

Name: TCP Wrappers 7.4

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: TCP Wrappers monitors and filters incoming requests for network services. Services monitored include SYSTAT, FINGER, FTP, TELNET, RLOGIN, RSH, EXEC, TFTP, and TALK.

Purpose: TCP Wrappers reports the name of the host and the client service requested. It can be set to restrict what systems can connect to which network services. It offers additional protection against systems pretending to have another hosts name or address.

Usage: TCP Wrappers can be used by the security administrators to monitor or control requests for network services.

References: Online documentation.

4.3.8.6 Tripwire 1.0.0.1

Name: Tripwire 1.2

Platform: Solaris 2.4 and 2.5.1
HP-UX 9.07

Description: Tripwire monitors changes in the file system.

Purpose: Tripwire gives the security and system administrators the capability to confidently determine system integrity. Tripwire provides 8 methods of generating an electronic signature for each file. If file contents are modified, the signature changes.

Usage: Tripwire is useful in detecting an intrusion and in determining the extent of corruption. It can provide positive proof of a break-in.

References: Online documentation.

4.4   DII COE Common Support Applications

Common Support Applications are provided to extend Kernel functionality to the user. These applications consist of tools and utilities that are used directly by the users, or suites of tools which form an actual end-user application system. Elements of these suites may be used by developers in developing new applications.

4.4.1   Office Automation Services

Extended Office Automation Services that extend the basic desktop environment provided by the Kernel are not provided in the UNIX environment. Solaris-based systems can install the Windows® NT Office Automation Services on WABI equipped workstations.

4.4.1.1   Internet Relay Chatter Server

Name:	Internet Relay Chatter Server (IRCS) 1.0.0.1
Platform:	Solaris 2.4 HP-UX 9.07
Description:	IRCS is a COTS application providing server functionality for real-time interactive conferencing over the Internet.
Purpose:	TBD
Usage:	Allows users to communicate with each other in real-time using the Internet connection.
References:	Vendor documentation -- TBD.

4.4.1.2   Internet Relay Chatter Client

Name:	Internet Relay Chatter Client (IRCC) 1.0.0.2
Platform:	Solaris 2.4 HP-UX 9.07
Description:	IRCC provides client functionality for real-time interactive conferencing over the Internet.
Purpose:	TBD

Usage: Allows users to communicate with each other in real-time using the
Internet connection.

References: Vendor documentation -- TBD.

4.4.1.3   Mail Services 1.0.0.2

Name: Mail Services (MSVCS) 1.0.0.2

Platform: Solaris 2.4
HP-UX 9.07

Description: MSVCS is a set of tools providing electronic mail functions.

Purpose: TBD

Usage: Provides e-mail services to end-users.

References: Vendor documentation -- TBD.

4.4.2   Mapping, Charting, Geodesy & Imagery (MCG&I) Services

Name: Joint Mapping Toolkit (JMTK) 1.0.0.5

Platform: Solaris 2.4
HP-UX 9.07

Description: JMTK is a GOTS system providing objects and services to support
geospatial analysis, visual display mapping, geospatial database
management, and image preprocessing.

Purpose: JMTK provides common geospatial processing and data to all
mission applications and users.

Usage: JMTK provides visualization, analysis, and spatial database
management software capabilities for standard geographical
information types. Functional services include:

• Spatial Database Management,
• Visual,
• Analysis,
• Utilities,

- Local Image Manipulation,
- Overlay Manager, and
- Security, Access and Data Releasability.

References: DII COE JMTK Version 3.0 Developers Manual - Part 2 (Man Pages)

4.4.3   Messaging Services

Two versions of the Common Message Processor (CMP) are available. The difference is in the method used to save message traffic.

4.4.3.1   CMP 1.0.1.1 (w/Informix)

Name: Common Message Processor 1.2.1.1.

Platform: Solaris 2.5

Description: The CMP is a generic, table-driven United States Message Text Format (USMTF) processor that prepares, receives, analyzes, and validates USMTF messages. The message data are stored in Informix database tables.

Purpose: The CMP is the COE's message handling portion of the common support software suite.

Usage: The CMP converts message data into a format usable by the host application software from incoming messages. It also converts data in the format used by the host application software or databases into that of the USMTF or other message formats for out-bound messages. The system can accommodate all messages in the USMTF standards as well as the other agreed upon message standards. These may be customized to meet users' needs. The system also supports plain language address databases for message addresses.

References: Software User's Manual (SUM) for the DII COE CMP, Version 1.2 (29 August 1996),
Application Programming Interface (API) for the DII COE CMP, Version 1.2 on the Sun Solaris (28 May 1996), and
Software Requirements Specification (SRS) for the DII COE CMP, Version 1.2 on the Sun Solaris (10 October 1996).

4.4.3.2 CMP 1.0.2.0 (w/o Informix)

Name: Common Message Processor 1.2.2.2.

Platform: Solaris 2.5

Description: The CMP is a generic, table-driven United States Message Text Format (USMTF) processor that prepares, receives, analyzes, and validates USMTF messages. The message data are stored in standard UNIX files.

Purpose: The CMP is the COE's message handling portion of the common support software suite.

Usage: The CMP converts message data into a format usable by the host application software from incoming messages. It also converts data in the format used by the host application software or databases into that of the USMTF or other message formats for outbound messages. The system can accommodate all messages in the USMTF standards as well as the other agreed upon message standards. These may be customized to meet users' needs. The system also supports plain language address databases for message addresses.

References: Software User's Manual (SUM) for the DII COE CMP, Version 1.2 (29 August 1996),
Application Programming Interface (API) for the DII COE CMP, Version 1.2 on the Sun Solaris (28 May 1996), and
Software Requirements Specification (SRS) for the DII COE CMP, Version 1.2 on the Sun Solaris (10 October 1996).

4.4.4   Alerts Facilities (1.3.4.1)

Name: Alerts Software 1.3.4.1

Platform: Solaris 2.4

Description: The alerts service builds on the signal processing functions in the Kernel to manage tactical processing on a workstation or, with an alerts server, on a family of workstations.

Purpose: Alerts provides a standard way for user applications to initiate local and persistent alerts.

Usage: The alerts service defines alert and event message formats and controls the posting, reviewing, queuing, and dequeuing of alerts and events. Events can be generated by a process or generated by a system event. The alerts service provides:

- Capability to send and delete alerts; receives responses from alerts.
- Capability for receiving, displaying, and responding to alerts.
- Capability to enable or disable the ability to suspend alert notification, suspend or enable local alert notification, and enable or disable processing of non-critical alerts.

References: TBD

4.4.5 Correlation and Situation Display Services (Unified Build)

Unified Build is an application that provides for battlefield management. It provides a situation display and track correlation to the end user. From a developer's point of view, it consists of 5 API toolkits, one of which was discussed in the Communications Infrastructure Services sub-section.

4.4.5.1 Track Database Management

Name: Track Database Management (TDBM) Service Application

Platform: Solaris 2.4
HP UX 9.07

Description: The Track Database Manager (TDBM) is the component of DII COE that provides multi-source correlation and database management of tactical track data.

Purpose: The track database consists of data records that contain the identifying attributes and position histories of reported ships, submarines, aircraft, land units and other moving or fixed objects of interest in the world.

Usage: The track database can display an exercise environment, a real-world environment, or view both. It includes API's that relate to:

- database management,
- network database management, and
- correlation.

References: TDBM Service Application Programmers Interface (API) for the DII COE Software Development Environment.

4.4.5.2 Application/Tactical Decision Aids Toolkit

Name: Application/Tactical Decision Aids (TDA) Toolkit

Platform: Solaris 2.4
HP UX 9.07

Description: The Application/TDA Toolkit provides routines that are available in COE's Software Development Environment (SDE) libraries, which are used as general purpose routines in building applications.

Purpose: The Application/TDA Toolkit comprises routines for data checking, data conversion, data file access, math calculations, PIM track manipulations, and general purpose utilities.

Usage: The Application/TDA Toolkit encompasses a collection of functions that facilitate the development of DII COE-based applications. The functions can be categorized as follows:

- Data File Access Routines
- Data Checking Routines
- Data Conversion Routines
- General Purpose Routines
- Math Calculation Routines
- PIM Track Routines
- String Manipulation Routines.

References: Application/TDA Toolkit Application Programmers Interface (API) for the DII COE Software Development Environment

4.4.5.3 CHART

Name: CHART (Cartographer)

Platform: Solaris 2.4
HP UX 9.07

Description: The chart system is based on a client server architecture in which the server provides geographic display capabilities to its clients. Cartographer manages the geographic display windows for a number of machines on the Local Area Network (LAN).

Purpose: CHART (or Cartographer) provides a powerful, integrated environment for developing network-based Geographic Information System (GIS) applications. These applications are put to use in such diverse areas as military command and control, automated freight tracking—even marine science, where routes of migrating Gray Whales are monitored.

Usage: To create and manage the geographic display windows at the request of client-application programs. This includes:

- Map Products
- Map Features
- Client Objects
- Cartographer/Client Communication
- Cartographer Utilities.

References: CHART Application Programmers Interface (API) for the DII COE Software Development Environment

4.4.5.4   HELP Service

Name: Help Service (HelpServ)

Platform: Solaris 2.4
HP UX 9.07

Description: The Help Service, or HelpServ by name, is a Motif application that is loaded into the background when a DII COE-based system starts. The Help Service uses navigation and help files which are ASCII readable and can be edited with any conventional text-based word processor.

Purpose: To remove responsibility for handling Help files from the application to a centralized server application that directs all aspects of display and user interaction. This allows incremental changes to the on-line Help system without requiring any code changes in existing applications.

Usage: HelpServ creates and manages its windows in off-screen memory, and maps the proper windows to the display when requested by a client application.

References: Help Service Application Programmers Interface (API) for the DII COE Software Development Environment

This page intentionally left blank

5. DII COE Microsoft NT Systems

5.1   COE Kernel (NT - Native)

COE Version: 3.0.0.3

Platform: Microsoft -- NT Version 3.5.1

Description: The DII COE Kernel for NT is the minimum set of software required for the operation of any DII COE compliant workstation for NT 3.5.1 regardless of how the workstation is used.

Purpose: NT includes both GUI and programmatic access to all DII COE kernel services such as services for traditional operating system capabilities, windowing, system administration, security, and printing. There are other services documented in NT that are not part of the DII COE.

Usage: The NT native services are presented in the NT 3.5.1 documentation and GUI guide. NT is based on Microsoft's Windows Open Services Architecture (WOSA) which provides a single, open-ended interface to enterprise computing environments. NT hides programming complexities from users and application developers by providing high-level interfaces such ODBC, MAPI, LSAPI, SNA API, Windows Sockets, and RPC. NT supports the Component Object Model (COM) and Object Linking and Embedding (OLE) 2.0 specifications that provide various ways to integrate application components, including features such as visual editing, 'Drag and Drop' between applications, OLE Automation, and structured storage for objects. NT provides both GUI and programmatic access to its underlying features. GUI access is provided with a built-in windowing capability similar to Microsoft Windows 3.1. Programmatic access is provided by the Win32 API.

References: Vendor documentation

5.1.1   Operating System

The operating system delivered with the COE consists of the basic, vendor-supplied operating system and any operating system patches required to operate the COE.

Platform: NT 3.5.1

Description: POSIX-compliant operating system with a host of built-in features including GUI, networking and workgroup support, administration

tools, security, file sharing and print sharing capabilities, data protection, Remote Access Service (RAS), file backup, and command prompt.

Purpose: Provides basic and advanced operating system features in a single system. Allows a server to provide multiple services for various purposes, e.g., communications server, application server, database server, and file server.

Usage: The operating system can be accessed via the built-in GUI or via the command prompt. There are a common set of POSIX calls which are listed in Appendix D. These calls enable the programmer to:

- create, execute, and terminate processes; communicate between processes; and coordinate process operations,
- create, delete and process files and directories,
- activate a process on a timed basis or in response to an external event,
- manage configurable system parameters, and
- access basic security functions inherent in the operating system.

References: Vendor documentation.
Publications for NT 3.5.1 are available in bookstores.
Multiple Internet Websites for acquiring Information:
　　　http://www.microsoft.com/
　　　http://moli.microsoft.com/
　　　http://windows-nt.com/
　　　http://www.bhs.com/

5.1.2　Windowing

Name: Program Manager

Platform: NT 3.5.1

Description: GUI shell similar to the GUI shell in Windows 3.1.

Purpose: Provides GUI access to system functions and applications.

Usage: The Program Manager is automatically run when you boot NT and is the GUI shell which is accessed with either mouse or keyboard input, similar to Windows 3.1.

References: Vendor documentation.

Multiple Internet Websites for acquiring Information:
http://moli.microsoft.com/
http://windows-nt.com/

5.1.3   System Administration

Name: (built-in system administration features)

Platform: NT 3.5.11

Description: NT provides built-in and comprehensive system administration tools for all system features. Some NT programs are multipurpose, i.e., they may act as both a system utility and a system administration tool. For instance, the File Manager is used to copy files as well as to administer the sharing of system storage resources. System administration tools include File Manager, Control Panel, Print Manager, NT Setup, PIF Editor and all tools in the Administrative Tools program group and the Network Administration program group.

Purpose: To configure the operating system.

Usage: System administration tools are accessed via the GUI.

References: Vendor documentation.
Multiple Internet Websites for acquiring Information:
http://moli.microsoft.com/
http://windows-nt.com/

5.1.4   Security

Name: (built-in security features)

Platform: NT 3.5.1

Description: NT provides built-in security features. Some NT programs are multipurpose, i.e., they may act as both a system utility and a security administration tool. For instance, the File Manager is used to copy files as well as establishing permissions to protect storage resources. Security administration tools include File Manager, Print Manager, User Manager for Domains, and the User Profile Editor.

Purpose: To configure security settings.

Usage: Security administration tools are accessed via the GUI.

References: Vendor documentation.
Multiple Internet Websites for acquiring Information:
      http://moli.microsoft.com/
      http://windows-nt.com/

5.1.5   Print Services

Name: Print Manager

Platform: NT 3.5.1

Description: NT provides a built-in print manager.

Purpose: Connect to, configure and share printers; control printing of documents.

Usage: The Print Manager is accessed via the GUI.

References: Vendor documentation.
Multiple Internet Websites for acquiring Information:
      http://moli.microsoft.com/
      http://windows-nt.com/

5.1.6   COE Runtime Tools 1.0

Name: Runtime Tools 1.0

Platform: NT 3.5.1

Description: The runtime tools developed for the COE provide the basic method used to install segments in this environment.

Purpose: Used to install segments.

Usage: Although primarily designed to support the system administrator, the applications developer must use these basic tools in constructing any scripts run prior to or after the actual installation. COE Runtime Tools are normally executed on the command line or as a menu item in the system administrator application. Some of the COE Runtime Tools do provide functionality through APIs which are of general use. They allow an application to:

- solicit a yes/no type response from the user,
- display a window with a general message or a termination message,

- prompt the user for a text response, and
- prompt for passwords with a verify option.

References: *DII COE I&RTS* Version 2.0 - Appendix C.
DII COE Application Programmer Interface Reference Guide for
Windows NT (2.0.0.1).
Vendor documentation.
Multiple Internet Websites for acquiring Information:
http://moli.microsoft.com/
http://windows-nt.com/

5.1.7   Distributed Computing Services - Transarc DCE Client

Name: Transarc DCE 1.03 (Client)

Platform: NT 3.5.1

Description: Open systems solution to distributed computing. The DCE is
composed of a series of services. The NT platform supports only
DCE client services.

Purpose: DCE provides a communications environment to hide the
complexity of network communications from the application
developer. This is accomplished by placing layers of services on top
of the network services. The DCE Core services support:

- programs performing many actions concurrently (Threads),
- direct calls to procedures on remote systems (Remote
Procedure Calls or RPC),
- logically naming objects within a DCE cell (Cell Directory
Service or CDS) or throughout a network (Global Directory
Service or GDS), and
- authentication of users and determination of privledges
accross machines.

Usage: DCE simplifies the process of developing distributed applications.
DCE Threads are based on the POSIX threading standard and do
not introduce any new complexity into the process. RPC calls
eliminate explicit programming of network communications and
mask differences in data representation across platforms. The
directory services allow resources to be identified by name only,
without reference to physical location.

References: Vendor documentation; on-line information on the Internet:
- http://www.transarc.com (Transarc's homepage)

• http://www.osf.org/dce (Open Systems Foundation)

5.2   DII COE Developer's Tools

The DII COE Developer' S Toolkit helps programmers with the creation, installation, testing and management segments.

5.2.1   COE Developer's Toolkit

Name: Developer' s Toolkit

Platform: Windows NT 3.5.1

Description: The DII COE Tools allow the programmer to create, install and test segments.

Purpose: The toolkit aids in the development of segments and the standardization of procedures for loading segments.

Usage: The Developer's Toolkit will be used by programmers and software integrators to:

• compute space required by a segment,
• test a segment,
• write segments to an installation medium or package segments,
• temporarily install a segment,
• remove a segment,
• timestamp a descriptor,
• validate a segment, and
• update segment data in the descriptor.

References: *DII COE Programmer' s Manual*,
DII COE I&RTS,
DII COE Programmer's Guide, Version Description Document (VDD), and
 DII COE Developer' s Toolkit (Windows NT).

5.3   COE Infrastructure Tools (Native to NT Only)

The DII COE Infrastructure Tools for NT are not provided in this version of DII COE. Updated information will be provided in later versions. Currently the Infrastructure Tools categories are:

• system management,
• network management,

- print management,
- communications,
- data management,
- distributed computing service,
- generalized utilities, and
- security utilities.

These services are partially provided within the DII COE. However, some equivalent tools exist in the Native NT environment which are not components of the DII COE.

5.3.1   Systems Management

Name:	Not Available in this release -- N/A
Platform:	N/A
Description:	N/A
Purpose:	N/A
Usage:	N/A
References:	N/A

5.3.2   Network Management

Name:	Not Available in this release -- N/A
Platform:	N/A
Description:	N/A
Purpose:	N/A
Usage:	N/A
References:	N/A

5.3.3   Print Management

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.4   Communications

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.5   Data Management

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6   Distributed Computing Services

The Distributed Computing Services (DCS) consist of Distributed Computing Environment (DCE) server and client components. The client component is available on the NT and is supported in DII COE. However the server components are not available in this release but a place holder for each of the future components are provided.

5.3.6.1   DCE Server 1.1

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6.2   DCE DFS 1.1

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6.3   DCE Cell Manager 1.1

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6.4   Netscape Browser 2.0

Name: Netscape Browser 2.01

Platform: Windows NT 3.5.1

Description: Netscape Navigator is a COTS graphical client application for accessing the WWW and other HTTP networks.

Purpose: Netscape Navigator provides graphical tools for accessing the WWW and searching, reviewing, and retrieving information from available sources, which include government sites, commercial sites, and educational sites, to name a few. It also supports electronic mail and access to newsgroups.

Usage: Netscape Navigator is used to access WWW sites on the SIPRNet using Netscape Navigator capabilities. This allows access to the SIPRNet Web, SIPRNet Newsgroups, and SIPRNet e-mail. Installation and usage require Netscape Navigator 2.01 licenses.

References: Netscape Communications Corp.
Netscape on-line documentation (http:/www.netscape.com)
DII System Administrator' s Manual

5.3.6.5   Netscape NewsServer 2.0

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6.6   Netscape WebServer 2.0

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.6.7   News Make Group 1.0.0.1

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.7   General Utilities

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.3.8   Security Utilities

     Name: Not Available in this release -- N/A

     Platform: N/A

     Description: N/A

     Purpose: N/A

     Usage: N/A

     References: N/A

5.4   DII COE Common Support Applications

The DII COE common support applications for NT consist of:

- Office Automation,
- MCG&I Services,
- Messaging Services,
- Alerts,
- Correlations, and
- Situation Display Services.

The NT has limited support for DII COE specific segments. The supported Office Automation segments are as follows:

- Internet Relay Chat (IRC) Client,
- Exchange,
- PowerPoint,
- Word, and
- Excel.

Except for the IRC segment that is DII COE all other segments are native mode NT products.

5.4.1   Office Automation Services

The office automation services for NT consist of the Internet Relay Chat for Client and the native Windows NT Office functions of: Exchange, PowerPoint, Word for Windows, and Excel.

5.4.1.1   IRC Client 1.0.0.1

Name:	Internet Relay Chatter Client (IRCC)
Platform:	NT 3.5.1
Description:	IRCC is the client component of a real-time conferencing tool. Messages input to a conference are made visible to other participants in the conference within seconds.
Purpose:	Information not available for this release -- TBD
Usage:	Information not available for this release -- TBD
References:	Information not available for this release -- TBD

5.4.1.2   Exchange

Name:	Microsoft Exchange
Platform:	NT 3.5.1
Description:	Distributed Electronic Mail for NT platform. Additional information TBD.
Purpose:	Electronic Mail Exchange within a LAN or across WAN. Additional information TBD.
Usage:	Provides functionality to create interpersonal messages for transfer across organizations. Additional information TBD.
References:	Vendor documentation. Multiple Internet websites for acquiring information: http://moli.microsoft.com/ http://windows-nt.com/

5.4.1.3   PowerPoint

Name:	Microsoft PowerPoint 4.0
Platform:	NT 3.5.1
Description:	Presentation Graphics Program
Purpose:	Develop content for presentations and rehearse the presentation.

Usage:	PowerPoint is accessed via the GUI.
References:	Vendor documentation.
	Multiple Internet Websites for acquiring Information:
	http://moli.microsoft.com/
	http://windows-nt.com/

5.4.1.4   Word (for Windows)

Name:	Microsoft Word for Windows.
Platform:	NT 3.5.1
Description:	Word processing program.
Purpose:	Write, edit, print documents and forms using GUI based editor.
Usage:	Word for Windows is accessed via the GUI.
References:	Vendor documentation.
	Multiple Internet websites for acquiring information:
	http://moli.microsoft.com/
	http://windows-nt.com/

5.4.1.5   Excel

Name:	Microsoft Excel Version TBD.
Platform:	NT 3.51
Description:	Spreadsheet program.
Purpose:	To store and manipulate tabular data, producing output from the data, e.g., textual displays of totals, graphs.
Usage:	Excel is accessed via the GUI.
References:	Vendor documentation provided by suppliers.
	Multiple Internet Websites for acquiring Information:
	http://moli.microsoft.com/
	http://windows-nt.com/

5.4.2 Mapping, Charting, Geodesy & Imagery (MCG&I) Services

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.4.3 Messaging Services

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.4.4 Alerts

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

5.4.5   Correlation Situation Display Services (Unified Build)

Name: Not Available in this release -- N/A

Platform: N/A

Description: N/A

Purpose: N/A

Usage: N/A

References: N/A

Appendix A: Acronyms

ANSI	American National Standards Institute
API	Application Program Interface
ASCII	American National Code for Information Interchange
CASE	Computer-Aided Software Engineering
CDE	Common Desktop Environment
CDS	Cell Directory Service
CMP	Common Message Processor
COE	Common Operating Environment
COM	Component Object Model
COTS	Commercial Off The Shelf
CSE	Client Server Environment
CSTC	Computer Security Technology Center
DCE	Distributed Computing Environment
DCS	Distributed Computing Services
DFS	Distributer File Services
DII	Defense Information Infrastructure
DISA	Defense Information Systems Agency
DLL	Dynamic Link Library
DoD	Department of Defense
DODIIS	DoD Intelligence Information System
ECMA	European Computer Manufacturers' Association
ELINT	Electronic Intelligence
FAQ	Frequently Asked Questions
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol
GDS	Global Directory Service
GIF	Graphics Interchange Format
GIS	Geographic Information System
GOTS	Government Off The Shelf
GUI	Graphical User Interface
GUNZIP	Name of a decompression tool
GZIP	Name of a compression tool
HDBK	Abbreviation for DoD handbook
HP	Hewlett Packard
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol

I/O	Input/Output
I&RTS	Integration and Runtime Specification
IEC	International Electrotechnical Commission
IEEE	Institute for Electrical & Electronic Engineers
IRC	Internet Relay Chat
IRCC	Internet Relay Chatter Client
IRCS	Internet Relay Chatter Server
ISEE	Integrated Software Engineering Environment
ISO	International Standards Organization
JIEO	Joint Interoperability and Engineering Organization
JMTK	Joint Mapping Toolkit
JPEG	Joint Photographic Experts Group
JSTARS	Joint Surveillance Target Attack Radar System
LAN	Local Area Network
LSAPI	License Service API
LZW	Linpei-Ziv Welch (compression algorithm)
MAPI	Mail Application Programmer Interface (MS-Windows)
MCG&I	Mapping, Charting, Geodesy, and Imagery
MIB	Management Information Base
MIL	Military designation of documentation
MSVCS	Mail Services
MWM	Metal Working Machinery
	Motif Window Manager
NIS	Network Information Service
NIST	National Institute of Standards and Technology
NMF	Network Management Forum
NT	New Technology (MS-Windows)
ODBC	Open Database Connectivity
OLE	Object Linking & Embedding
OLTP	Online Transaction Processing
OSF	Open Software Foundation
	(DISA/JIEO Operation Support Facility)
PASC	Portable Application Standards Committee
PCTE	Portable Common Toolkit Environment
Perl	Practical Extraction and Report Language
PIF	Page Image Format
PIM	Position and Intended Movement
PKZIP	Name of a compression tool
POC	Point of Contact

POSC	Portable Operating System Committee
POSIX	Portable Operating System Interface for UNIX
PRM	Programmer's Reference Manual
RAS	Remote Access Service
RPC	Remote Procedure Calls
SA	System Administration
SAS	SPARC STORAGE Arrays
SDE	Software Development Environment
SIPRNet	Secure Internet Protocol Router Network
SNA	System Network Architecture
SMP	Symmetric Multi-Processor
SPI	Security Profile Inspector
SQL	Standard Query Language
SRS	Software Requirement Specification
SS	System Services
STD	Standard (use for DoD documentation)
TAC	Tactical Advanced Computer
TADIL	Tactical Digital Information Links
TAFIM	Department of Defense Technical Architecture Framework for Information Management
TBD	To Be Determined
TCIM	Tactical Communications Interface Module
TCL	Tool Command Language
TDA	Tactical Decision Aids
TDBM	Track Database Management
TED	TriTeal Enterprise Desktop
Tk	Toolkit
UB	Unified Build
UIL	User Interface Language
USMTF	United States Message Text Format
UX	UNIX
VDD	Version Description Document
WABI	Windows Application Binary Interface
WAN	Wide Area Networks
WEBSv	Netsite Web Server
WOSA	Window Open Services Architecture
WWW	World Wide Web
XAPIA	X.400 API Association

X11	X-Windows, Version 11
X11R5	X-Window, Version 11, Release 5
XM	Motif Toolkit
	Motif widgets
XPG4	X-Open Programming Guide
XT	X Toolkit Intrinsics

Appendix B: List of COE Tools and APIs

B.1   Description of Solaris and Hewlett Packard Tools

API NAME	Description
DII COE Toolkit - Ref. DII COE Version Description Document for the Kernel (HP and Solaris) DRAFT Version 3.0.0.3 *Also found in the DII COE Application Programmer Interface (API) Reference Guide (HP and Solaris) DRAFT Version 3.0.0.3	
*COEAskUser 1.0.0.6	Can be called from anywhere to display a message to the user.
*COEFindSeg 1.0.0.6	Searches a standard location for a given segment directory or a given segment name to see if it is on disk and can be read. If so, it returns the directory name, the segment name, the prefix, the type and, optionally, the attributes.
COEInstaller 1.0.0.8	Displays a list of segments which may be installed from tape, disk or other electronic media.
*COEInstError 1.0.0.6	Displays an error to the user.
*COEMsg 1.0.0.6	Displays a message to the user from within `PreInstall`, `PostInstall`, or `Deinstall` script.
*COEPrompt 1.0.0.6	Displays a message to the user and expects the user to enter a response from within a `PreInstall`, `PostInstall`, or `Deinstall script`.
*COEPromptPasswd 1.0.0.6	Displays a message to the user and waits for the suer to enter a password.
*COEUpdateHome 1.0.0.4	Updates the home environment within a script file to point to where a segment actually was installed.
DII COE Developer's Toolkit - Ref. DII COE Version Description Document for the Developer's Toolkit (HP and Solaris) DRAFT Version 3.0.0.3	
CalcSpace 1.0.0.4	Computes the space required by a segment and updates the hardware descriptor
CanInstall 1.0.0.6	Tests a segment to see if it can be installed.
ConvertSeg 1.0.0.7	Examines segment descriptors and converts them to the latest format.
MakeAttribs 1.0.0.7	Creates the descriptor file `FileAttribs`, and recursively traverses every subdirectory beneath the segment and creates a file with lines in the proper format.
MakeInstall 1.0.1.5	Writes one or more segments to an installation medium or packages the segments for distribution over the network.
TestInstall 1.0.0.7	Temporarily installs a segment that already resides on disk.
TestRemove 1.0.0.6	Removes a segment that was installed by TestInstall.
TimeStamp 1.0.0.6	Puts the current time and date into the `VERSION` descriptor.
VerifySeg 1.0.0.7	Validates that a segment conforms to the rules for defining a segment.
VerUpdate 1.0.1.5	Updates the segment version number, date and time in the `VERSION` descriptor.

API NAME	Description
DII COE Printer API Toolkit - Ref. DII COE Application Programmer Interface (API) Reference Guide, (HP and Solaris) DRAFT Version 3.0.0.3	
close_printer	Concludes a single print job and sends the data to the printer. If close_printer is not called, the print job will not be printed.
get_printer_descriptions	Retrieves the description of the default printer.
get_printer_name	Retrieves the name of the default printer.
get_printer_type	Retrieves the type of default printer. The type is either (1) "ASCII," (2) "HPGL," or (3) "PostScript".
open_printer	Establishes a print context, including the security level, line length, page length, line spacing, and indentation for the print job.
page_break	Indicates that subsequent data sent to a print job should begin on a new page.
VDirectPrintFile	Prints a file directly to the specified printer.
VDirectPrintMsg	Prints a text message directly to the COE default printer.
VPrintFile	Prints a file directly to an operator selected printer. The function retrieves all available printers on the network and displays them in a window for operator selection or cancellation.
VPrintMsg	Prints text message directly to an operator selected printer. The function retrieves all available printers on the network and displays them in a window for operator selection or cancellation.
write_printer	Writes a character string to the specified print job.
write_printer_array	Writes an array of strings, separated by line feed characters, to the specified print job.
EM_get_current_printer_desc	Returns the text description of the currently selected printer.
EM_get_current_printer_name	Returns the name of the currently selected printer.
EM_get_current_printer_type	Returns the type of default printer. The type is either "ASCII," "HPGL," or "PostScript".
DII COE Menu Executive API's - Ref. DII COE API Reference Guide (HP and Solaris) DRAFT Version 3.0.0.3	
MEAddMenus	Appends the menus defined in the menu file `filestr` to the menu bar `mb_num`.
MEChkSecurity	Sends a message to the Menu Executive service to check the current security settings from the Security Shell.
MECleanup	Destroys the specified application menu bar and frees all associated allocated memory without terminating the application.
MEClose	Logs out of the Menu Executive service designated by the fd.
MEDeleteMenus	Deletes the menus in the menu group specified by the id returned from MEAddMenus from the menu bar `mb_num` associated with socket fd.
MEExecProg	Sends the command `cmdstr` to the Menu Executive service to execute.

API NAME	Description
MEFlashIcon	Sends a command to the system menu bar connected to socket fd to flash the fixed icon with the specified item id.
MEGetEvent	Takes the MEEventPtr returned by MEGetNextEvent as an input and retrieves relevant data.
MEGetMBWidget	Returns the widget id of the menu bar specified by mb_num.
MEGetNextEvent	Waits to receive the next event from the socket fd.
MEIcon2Str	Retrieves the icon name string for the fixed icon with the specified id from the system menu bar connected to socket fd.
MELocalMenuBar	Creates a menu bar with specified height in the parent widget.
MEOpen	Opens a connection to the Menu Executive Service running on the display named by the input parameter.
MEStr2Icon	Retrieves the item id for the fixed icon with the specified name from the system menu bar connected to socket fd.
METextAlert	Sends the text string alertstr to the system menu bar.
DII COE User Profile Information API Toolkit - Ref. DII COE User Profiles Application Programming Interface, Version 3.0.0.2/Beta	
COE_add_user_data	Adds a new user record to the UserData table.
COE_add_user_reserved_data	Adds or modifies reserved field information in an existing UserData table entry.
COE_delete_user_data	Removes a user entry and all profile assigments to the user from the user profile database.
COE_delete_user_reserved_data	Deletes the specified reserved field(s) from the UserData table entry that matches UserId. No other fields in the record are affected.
COE_get_user_data	Returns information from the UserData table.
COE_get_user_id	Returns the suer's UNIX identification number from the given user login name.
COE_add_profile_data	Adds a new record to the ProfileData table.
COE_add_profile_reserved_data	Adds or modifies reserved field information in an existing ProfileData table entry.
COE_delete_profile_data	Removes a profile entry from the user profile database. It also removes any user-profile and profile-application assignments that reference this profile.
COE_delete_profile_reserved_data	Deletes the specified Reserved field(s) from the database entry that is defined by the ProfileName. No other fields in the record are affected.
COE_get_profile_data	Returns information from the ProfileData table.
COE_assign_user_to_profile	Assigns a profile to a user and saves the information to the user profile database. Verifies that the user and profile have been previously defined in the user database.
COE_deassign_user_from_profile	Removes the assignment of a profile to a user from the user profile database.
COE_get_user_profile_data	Returns user/profile assignment data.
COE_add_app_data	Enters new application data into the user profile database.

API NAME	Description
COE_delete_app_data	Deletes application data from the user profile database. Removes any profile-application assignments that reference this application and account Group.
COE_get_app_data	Returns application information from the AppData table.
COE_assign_app_to_profile	Assigns an application to a profile and saves the information to the user profile database. Verifies that the profile and the application have been previously defined in the user profile database.
COE_deassign_app_from_profile	Removes an application-to-profile assignment from the user profile database. Verifies that the profile and the application have been previously defined in the user profile database.
COE_get_profileapp_data	Returns profile/application assignment data.
COE_get_current_profiles	Returns the current user's selected profiles.
COE_set_current_profiles	Stores the current user's selected profiles.
COE_clear_profile_lock	Unlocks th e Profile identified by the ProfileName.
COE_set_profile_lock	Locks the Profile identified by ProfileName.
COE_query_profile_lock	Determines the current lock state of the profile identified by ProfileName.
COE_delete_all_profile_locks	Removes all profile locks defined within the specified scope (local or global).
COE_get_error_string	Returns a string defining an error code.
COE_to_fname	Takes in a string and returns a string with all non-alphanumeric characters replaced by underscores.
DII COE JMTK API Toolkit - Ref. DII COE JMTK Version 3.0 Developers Manual - Part 2 (Man Pages)	
Symbol Manipulation APIs	Manipulates symbology and sets attributes. (17 APIs, Legacy code)
Drawable Display Objects APIs	Draws basic geometric shapes (lines, curves, polygons, ellipses), tests, and bitmaps on the display. (25 APIs, Legacy code)
Display Settings APIs	Sets the display attributes (brightness, blinking, etc.) for display objects. Includes standard textual attributes. (15 APIs, Legacy code)
Display Features APIs	Hides of shows (unhides) selected display objects. (23 APIs, Legacy code)
Edit Features APIs	Moves an object from a window, modifies one or a list of feature attributes on a map, and removes objects from a list or map products or features from the Draw Module or geographic display. (12 APIs, Legacy code)
Data Transformations APIs	Performs a standard set of unit conversions. (20 APIs. Legacy code)
Windows APIs	Provides interface to the Chart Manager and to various Windows from disparate Chart Clients. (25 APIs, Legacy code)
Display View APIs	Displays various views of the current map and its display boundaries and width along with changing the mode of the cursor. (6 APIs, Legacy code)
Spatial Database Retrieval APIs	Procedures regarding the Spatial Data Base to access data, manage connections, define geographic areas of interest, retrieve functions, and save matrix data. (17 APIs, Legacy code (4))
Display Utilities APIs	Performs various utility functions including animation. (15 APIs, Legacy code)

API NAME	Description
Coordinate Transformations APIs	Converts between various mapping coordinate systems. (15 APIs, Legacy code)
Symbol Library APIs	Adds to and lists elements in the symbol library. (11 APIs, Legacy code)
Error APIs	Processes errors within the JMTK. (22 APIs, Legacy code)
Display Query APIs	Extracts information about the display or objects in the display, including range and bearing. (11 APIs, Legacy code)
Symbol APIs	Chart Manager category reference routines and option parsing routines. (21 APIs, Legacy code)
Windows - Communication APIs	Performs functions such as opening and closing a communications channel to the Chart Manager, notifying Chart Manager of various Draw Module activities, flushing the Draw Module request queue, and shutting down the Chart Manager. (24 APIs, Legacy code)
Memory Manager APIs	Chart Manager memory allocation utilities and object search utilities. (6 APIs, Legacy code)
DII COE Toolkit - Ref. Application/TDA Toolkit Application Programmer's Interface (API) for the DII COE Software Development Environment (SDE)	
Data File Access	Provides the developer with high-level functions for accessing: Four Whiskey files, Overlay files, PIM tracks, Radar filesm Screen Kilo files, Sites Information, and Unit TIB files. Routines are also included to select individual records from each data file for calculations. (20 APIs)
Data Checking	Checks the validity of an operator's entry for fields such as latitude, longitude, time and other special formats. (8 APIs)
Data Conversion	converts from xy and xyz coordinate systems to lat/lng coordinate systems and back. Contains routines for ASCII-BAUDOT conversion and time conversion. (29 APIs)
General Purpose	A set of commonly used routines which each and every programmer will need access to from time to time, yet they do not fall into a specific category. (12 APIs)
Math Calculation	Consists of both range/bearing algorithms and track intercept algorithms. (9 APIs)
PIM Track	Converts a PIM track into a Track and a Track into a PIM track, as well as routines to get a PIM track's latitude and longitude. (4 APIs)
String Manipulation	Consists of general purpose string functions for filling strings, trimming strings, and finding fields within strings. (21 APIs)
DII COE Toolkit - COMMS Service Application Programmer's Interface (API) for the DII COE Software Development Environment (SDE)	
Input Communications	Processes incoming formatted ASCII messages from diverse sources. Includes creation of one or more Incoming Message Logs (ILOGs). Allows client applications to access, modify, and obtain notification of certain events from the ILOGs

API NAME	Description
Output Communicaitons	Processes outgoing formatted ASCII messages. Includes auto-forwarding performed by Mps, automatic track broadcasts, operator- created messages, and application-generated messages. Includes creation of one or more Outgoing Message Logs (OLOGs). Allows client applications to access, modify, and obtain notification of certain events from the OLOG.
DII COE Toolkit - TDBM Service Application Programmer's Interface (API) for the DII COE Software Development Environment (SDE)	
Database Management	Manages the track database to include the following abilities: create, edit, delete, merge, associate, disassociate track records; reprocess ambiguity records; compare tracks; and, create track summaries.
Network Database Management	Provides a consistent representation of the central DII COE-provided track database to all workstations in the local area network. Exports data record updates to each workstation so that applications have local copies of the track database. Requests notification of database changes.
Correlation	Determines assignment in the track database. Determines whether an incoming contact report provides data for either an existing track or a new track.
DII COE Toolkit - HELP Service Application Programmer's Interface (API) for the DII COE Software Development Environment (SDE)	
HelpServ	Creates and manages help topic windows in off-screen memory, and maps the proper windows to the display when requested by a Client application. Takes the form of a hierarchical file reader, with Help topics organized like an upside-down tree. At the top level are the general topics, which lead to sub-topics, and so on. Eventually a separate window displays the information the user desires. The interface allows the user to navigate up and down through the hierarchy, and to go to the top-level topics at any time.
DII COE Toolkit - CHART Application Programmer's Interface (API) for the DII COE Software Development Environment (SDE)	
Cartographer server	Creates and manages the geographic display windows at the request of client-application programs. Specifically, Cartographer Server manages the plotting of: Map Products, Map Features, andClient Objects.
Client applications	Manages geographic display windows. Creates, manages and destroys graphical objects in Cartographer's display list.
DII COE Toolkit - Version Description Document for the Remote Segment Installer Segment DRAFT Version 2.0.0.0	
COERmtInstall	Transfers one or more segments from the source site across the network to the target site. The selected segment(s) either can be loaded on the network installation server or installed directly on the target machine.
DII COE CMP API Toolkit - DII COE CMP Application Programer Interface (API) V 1.2.1.1/Solaris 2.4) - **Solaris Only**	
Inbound Message Processing	Handles all functions of processing of incoming messages to include: receipt, routing, storage, profile generation, validation, parsing, and delivering of incoming messages. (7 APIs)
Outbound Message Generation	Creates, edits, and releases outgoing messages. (5 APIs)

API NAME	Description
Support Services	Controls configuration parameters, monitors errors, monitors and validates security functions, normalizes messages, translates Variable Message Format (VMF), operates and complies with standards, sections and reassembles transmissions, annotates base messages, retrieves and retransmits messages, and allows retrospective seaches. (14 APIs)
Operational Journal	Lists all messages entering or leaving the message processing block. (17 APIs)

B.2 Description of Windows NT Tools

API NAME	TYPE	USE*	Description
DII COE Toolkit - Ref. DII COE Version Description Document for the Kernel (Windows NT) DRAFT Version 3.0.0.3			
COEInstaller 1.0.0.8	Menu Command	R	Supports the segment installation process. Displays a list of segments that may be installed from floppy diskettes.
DII COE Developer's Toolkit - Ref. DII COE Version Description Document for the Developer's Toolkit (Windows NT) DRAFT Version 3.0.0.3			
CalcSpace 1.0.0.4	Menu Command	D	Computes the space required for the segment specified and updates the hardware descriptor.
CanInstall 1.0.0.6	Menu Command	D	Tests a segment to see if it can be installed, whcih means that all required segments must already be on the disk and the disk cannot have nay conflicting segments.
MakeInstall 1.0.1.5	Menu Command	D	Writes one or more segments to an installation medium or packages the segments for distribution over the network..
TestInstall 1.0.0.7	Menu Command	D	Temporarily installs a segment that already resides on a disk.
TestRemove 1.0.0.6	Menu Command	D	Removes a segment that was installed by TestInstall.
TimeStamp 1.0.0.6	Menu Command	D	Puts the current time and date into the VERSION descriptor.
VerifySeg 1.0.0.7	Menu Command	D	Validates that a segment conforms to the rules for defining a segment.
VerUpdate 1.0.1.5	Menu Command	D	Updates the segment version number, date, and time in the VERSION descriptor.
DII COE Toolkit - Ref. DII COE Application Programmer Interface (API) Reference Guide (Windows NT) DRAFT Version 3.0.0.3			
COEAskUser	Code	R	Creates an interface that prompts the user for a question and gives a choice of Yes or No buttons to select. The message question and the answer button labels can be assigned by the user.
COEFindSeg	Code	R	Searches a given segment directory for a given segment name to see if it is on disk and can be read. Returns the segment directory, name, prefix, type and attribute.

API NAME	TYPE	USE*	Description
COEInstError	Code	R	Creates an interface that displays an installation error message.
COEMsg	Code	R	Creates an interface that displays a message.
COEPrompt	Code	R	Displays a prompt and a text area to accept user input.
COEPromptPasswd	Code	R	Displays an optional `Password` prompt message and an option `Verify` prompt.

This page intentionally left blank

Appendix C: List of POSIX Function Calls

Process Control

_exit()	- terminate a process after flushing any pending output
execle()	- provides an interface to the execve call with know arguments
execve()	- transfer to entry point of new executable - vector of arguments
fork()	- create a new process
getpgrp()	- get process group of specified process
getpid()	- get process identification
kill()	- send signal to a process
pause()	- stop until signal
pipe()	- create an interprocess channel
setpgid()	- join or create new process group
setsid()	- create session and set process group ID
sleep()	- suspend execution for interval
wait()	- wait for process to terminate
waitpid()	- wait for specific child process
tcgetpgrp()	- get foreground process group ID
tcsetpgrp()	- set foreground process group ID
times()	- get process times
uname()	- get name of current system

File Operations

access()	- determines the accessibility of a file
chdir()	- change working directory
chmod()	- change mode of file
chown()	- change owner and group of a file
close()	- delete a file descriptor
creat()	- create a new file
dup2()	- duplicate an open file descriptor, returning a new descriptor
dup()	- duplicate an open file descriptor
fcntl()	- test, set or clear read or write locked file
fstat()	- get file status
link()	- link to a file
lseek()	- move read or write pointer associated with a file or open device
mkdir()	- make a directory file
mkfifo()	- make a FIFO special file
open()	- open file for read or write and return descriptor
pathconf()	- get configurable pathname variables
read()	- read from a file
rename()	- change the name of a file

rmdir()	- remove a directory file
stat()	- get file status
tcflush()	- discard data written but not transmitted or received but not read
tcdrain()	- wait until all output written to the *fildes* object transmitted
tcflow()	- suspend transmission or reception of dataon the *fildes* object
tcgetattr()	- get parameters associated with *fildes* - store in *termios* structure
umask()	- set file creation mask
unlink()	- remove directory entry
ustat()	- get file system statistics
utime()	- set file times
write()	- write on a file

Signal Processing

alarm()	- schedule signal after specified time
sigaction()	- software signal facilities
sigaddset()	- add specified signal from specified signal set
sigdelset()	- delete specified signal from specified signal set
sigemptyset()	- initializes signal set so that all signals are excluded
sigfillset()	- initializes signal set so that all signals are included
sigismember()	- tests whether signal is member of signal set
sigpending()	- examine pending signals
sigprocmask()	- examine and change blocked signals
sigsuspend()	- wait for signal

System Parameters

cfgetispeed()	- returns the input baud rate stored in the *termios* structure
cfgetospeed()	- returns the output baud rate stored in the *termios* structure
cfsetispeed()	- sets the input baud rate stored in the *termios* structure
cfsetospeed()	- sets the output baud rate stored in the *termios* structure
sysconf()	- get configurable system variables
tcsendbreak()	- cause transmission of a stream of zero valued bits
tcsetattr()	- set parameters associated with terminal from *termios* structure
time()	- get date and time

Group and User IDs

getegid()	- get effective group ID
geteuid()	- get effective user ID
getgid()	- get real group ID
getgroups()	- get group access list
getuid()	- get user identity
setgid()	- set real and effective group ID
setuid()	- set user and group ID

Appendix D: Netscape Notes

NOTE: These Release Notes are in-lieu of other documentation (i.e., VDD and IP).

D.l   Netscape 2.0 for Window

The Netscape Release Notes downloaded from Netscape Navigator can be found in the Enclosures attachments to the *DII COE Programmer's Reference Manual.* The release for UNIX is 1.12, while the release for Windows is 2.0. This Netscape Notes appendix contains additional Netscape insatallation information.

Upgrading from Version 1.0:

> If you are upgrading directly to Netscape Navigator 1.12 from Version 1.0 or 1.0N, you need to upgrade your ~/ .MCOM-* configuration files by hand. You don't need to do this if you are upgrading from Version 1.1.

> While running the Version 1.0 Navigator, change you cach directory from ~/ .MCOM-cache to ~/ .netscape-cache. Exit the Navigator and delete your old ~/ .MCOM-cache directory. You can then rename the rest of the configuration files:

~/ .MCOM-prefereces	->	~/ .netscape-preferences
~/ .MCOM-bookmarks.html	->	~/ .netscape-bookmarks.html
~/ .MCOM-global-history	->	~/ .netscape-history
~/ .MCOM-HTTP-cookie-file	->	~/ .netscape-cookies

AIX: Netscape requires AIX 3.2.5 with the X11R5 and Motif 1.2 libraries. If you are running an older version, you will get undefined symbols at run-time; unfortunately, the only solution is to upgrade to a more recent version of the OS. We are told (but have not verified) that this executable works on AIX version 4.

SunOS 4.1:   The SunOS 4.1 package comes with two executables: netscape and netscape_dns. This is because Suns use two different, incompatible methods of resolving host names (Domain Name Service, and Network Information Service, formerly known as Yellow Pages.) A given site only need install one of these executables.

> It is usually possible to configure your YP/NIS server to consult a DNS server for resolution of hosts not in the YP maps. Consult your system administrator for details.

SunOS 4.1, Linux, and BSDI:

These distributions also includes a directory called "nls". This directory is a standard   part of the MIT X11R5 distribution, but is not included with X11R4- or X11R6-based systems (such as OpenWindows 3.0 or earlier, and XFree86 3.1 or later.)

On these systems, Netscape has been linked against X11R5 (because Motif 1.2.4 has some bad bugs in conjunction with X11R6.) Unfortunately, X11R5 has one rather serious bug, which is that if this "nls" directory does not exist, the program will dump core any time you try to copy or paste to or from a text field!

So, if you don't have the "nls' directory on your system, you will need to install it first. Here is where Netscape looks for it (these default pathnames are hardcoded into the executable):

> SunOS 4.1   /usr/lib/X11/nls/
> Linux: /usr/X386/lib/X11/nls
>
> BSDI: /usr/X11/lib/X11/nls

If you choose not to create the directory there, then you must set the $XNLSPATH environment variable to the directory where you did install it.

NetBSD, FreeBSD:

We have been told (but have not verified) that the BSDI binaries will work on x86 systems running NetBSD 1.0 or FreeBSD 2.0 (but not FreeBSD 1.1.5.1) file, many warnings about "unknown keysyms" will be generated when the program starts up, and most keyboard equivalents won't work. This is a general problem with running Motif programs on systems not configured for Motif, and so will be necessary on most Sun systems.

This file is included with all packages because some systems have an older version of this file, so you may still get some warning. The XKeysymDB file normally goes in /usr/lib/X11/SkeysymDB or /usr/openwin/lib/XKeysymDB, but you can override that with $XKEYSYMDB.

Also included with all distributions is a file called Netscape.ad, which lists the default resources which are built in to the program. It is not necessary to install this; it is provided for informational purposes. See the comment at the top of the file for more information.

If you get a "Cannot locate host" dialog at startup, this is a sign of problems related to name resolution. If you're on a Sun, see the comments above about the two executables.

If you get a "Cannot connect to host"dialog at startup, it could mean that you are behind a firewall, and need to tell Netscape about your SOCKS server. See the Preferences dialog under the Options menu, or consult your system administrator.

Please read the relaease notes under "Help -> Telease Notes". This document is updated as problems are found, so please check it before reporting a bug.

To unpack a compressed tar file into the current directory, use some variation of the following command:

     zcat the-file.tar.z l tar -vxf -

And remember, it's spelled N-e-t-s-c-a-p-e, but it's pronounced "Mozilla."

= =
= = =

BY CLICKING ON THE "ACCEPT" BUTTON, YOU ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE "DO NOT ACCEPT" BUTTON AND THE INSTALLATION PROCESS WILL NOT CONTINUE.

NETSCAPE NAVIGATOR END USER LICENSE AGREEMENT REDISTRIBUTION NOT PERMITTED

This License Agreement has 3 parts. Part I applies if you have not purchased a license to the software. Part II applies if you have purchased a license to the software. Part III applies to all license grants. If you initially
acquired a copy of the software without purchasing a license and you wish to purchase a license, contact Netscape Communications Corporation ("Netscape") on the Internet at http://www/netscape.com.

PART I -- TERMS APPLICABLE WHEN LICENSE FEES NOT (YET) PAID GRANT.

Netscape hereby grants you a non-exclusive license to use its accompanying software ("Software") free of charge if (a) you are a student, faculty member or staff member of an educational institution (K-12, junior college, college or library) or an employee of a charitable non-profit organization; or (b) your use of the Software is for the purpose of evaluating whether

to purchase an ongoing license tp tje Software. The evaluation period for use by or on behalf of a commercial entity is limited to 90 days; evaluation use by others is not subject to this restriction. Government agencies (other than public libraries) are not considered educational or charitable.


```
*************************************************
*  SegInfo
*************************************************


Security
UNCLASS

SegType
COTS

Direct
$ROOT:PostInstall

FilesList
$PATH: /h/COTS
$DIRS
WESSr
$PATH:  /h/COTS/WEBBr
$DIRS
Profiles
SegDescrip
bin
bitmap
data
install_utilities
lib
man
progs
$PATH: /h/COTS/WEBBr/Profiles
$FILES
LaunchDesc.WEBBr
LaunchList.Webbr
$PATH:  /h/COTS/WEBBr/SegDescrip
$FILES:
DEINSTALL
PostInstall
PostInstall.orgname
Profiles.WEBBr
```

ReleaseNotes
SegInfo
SegName
VERSION
Validated
$PATH: /h/COTS/WEBBr/bin
$FILES
gunzip
hot-conver.sh
netscape
netscape.init
$PATH: /h/COTS/WEBBr/bitmap
$FILES
webbr.img
$PATH: /h/COTS/WEBBr/data
$DIRS
app-defaults
$PATH: /h/COTS/WEBBr/data/app-defaults
$FILES
netscape.ad
$PATH: /h/COTS/WEBBr/install_utilities
$FILES
OS_fnxns.ksh
PostInstall_Xcaller.ksh*
PostInstall_fnxns.ksh*
PostInstall_msg.ksh*
LICENSE
README
XKeysymDB
Utilcap
mime.type
skel.netscape-preferences
$PATH: /h/COTS/WEBBr/man
$FILES
learn.html
$PATH: /h/COTS/WEBBr/progs
$FILES
gunzip
hot-convert.sh
netscape
netscape.init

Hardware
SCPU:SOL

$DISK:2000
$MEMORY:5000

D.2   Netscape 1.12 For UNIX

NETSCAPE NAVIGATOR 1.12 FOR SUN

You are using Netscape 1.12SUN for Solaris. These release notes cover

  * Installing the Netscape Navigator

  * Security Improvements

  * New Features

  * Upgrading from 1.0 or 1.1

  * Known Problems and Bugs

Netscape Servers Release Notes are also available for the Commerce, Communications, and Proxy servers.

INSTALLING THE NETSCAPE NAVIGATOR

The Netscape Navigator comes in two configurations, export and domestic. To install the Netscape Navigator:

  1. Insert the CD.

  2. Become superuser.

  3. Change to the Volume Manager directory for the Netscape Navigator:

   The directory name is one of the following, depending on which product
   you are installing:

     o netscape_navi_1_12_export
     o netscape_navi_1_12_domestic
     o netscape_comm_1_12
     o netscape_cmrc_1_12_export
     o netscape_cmrc_1_12_domestic
     o netscape_prox_1_12_export
     o netscape_prox_1_12_domestic

As an example:

```
    # cd /cdrom/netscape_navi_1_12_export
  or
    # cd /cdrom/netscape_navi_1_12_domestic
```

4. Install the navigator package:

Each CD-ROM contains both the SPARC and x86 versions of the software.

```
    # pkgadd -d sparc SUNWnsnav
  or
    # pkgadd -d x86   SUNWnsnav
```

The Navigator files will be installed into the following locations:

```
Navigator              /opt/SUNWweb/bin
Support Files          /opt/SUNWweb/etc
Default mailcap         /usr/local/lib/netscape
Locale-specific Files    /usr/openwin/lib/locale
```

5. Add /opt/SUNWweb/bin to your PATH in your .profile or .login

In addition, you may wish to manually add the directory to your PATH:

```
    # export PATH; PATH=${PATH}:/opt/SUNWweb/bin      sh & ksh
  or
    # setenv PATH ${PATH}:/opt/SUNWweb/bin            csh
```

NOTE: For more information about configuration, see the documenation for the Netscape Navigator.

SECURITY IMPROVEMENTS

 * Technical Background

Netscape Navigator uses random information to generate session
encryption keys of either 40 or 128 bits in length. The random
information is found through a variety of functions that look into a
user's machine for information about how many processes are running,

process ID numbers, the current time in microseconds, etc. Previous releases of Netscape Navigator were vulnerable because the size of random input was less than the size of the subsequent keys. This means that instead of searching through all the 2^128 possible keys by brute force, a potential intruder only had to search through a significantly smaller key space by brute force. This was a substantially easier problem to solve because it takes much less compute time and means 40-bit or 128-bit key strength is substantially reduced.

* Solution:

Netscape Navigator 1.22SUN (Windows), 1.12SUN (Solaris), 1.12ISUN (localized versions for Solaris) fixes the specific portion of the software where this vulnerability existed. We have significantly increased the amount of random information that cannot be discovered by external sources from approximately 30 bits to approximately 300 bits.

Netscape has greatly expanded the techniques and sources used to generate the random information. The number of unpredictable bits in the RNG makes it no longer the weak link in the chain.

* Recommendations to users of Netscape who rely on the security of their transactions:

  o If the attacker has physical access to your machine, security
    cannot be assured.

  o Netscape continues to point out that if unwanted agents can log
    into your machine, little can be secure. For secure servers, any
    insecure connection mechanism could be suspect. Multi-user UNIX
    platforms will not be as secure as single-user machines.
       Details: Some of the system specific information that is
       used in the seed generation is available to any user on that
       system.

  o To help mitigate this, the entire user environment is passed into
    the seed generation algorithm. A wary user can alter his user
    environment before running Netscape software.

  o If someone can get root (superuser) access to your machine, they
    can pretty much do anything.

o Netscape security could be weak if run on a platform emulator.
Use a version native to the platform on which you are running.
Details: Some of the usefulness of the seed generation
depends upon the unpredictability of the low order bits of
various clocks and timers. The clocks of many emulators may
have much less entropy than the actual builtin clocks.

o If you are running on a UNIX platform, make sure of the security
of your X server. This is also true if the X connection is not to
a local machine as all of the events and the data from the screen
read may be captured from the ethernet.

Details: If the attacker is monitoring your user input then
its randomness is not useful.

o It is better to perform some user action before connecting to a
secure site. This means you should not set your home page to a
secure site or launch your client from a command line to a secure

site.

Details: While navigating through menus or typing into
various form fields, the Navigator uses the unpredictability
of details of the user actions to increase the entropy of
the RNG state.

WHAT'S NEW?

1.12 to 1.12SUN Changes

* The default home page was changed from http://home.netscape.com/ to
  http://www.sun.com/
* An entry for www.sun.com was added to the Directory menu.
* The About menu choice (on the Go menu) reflects Sun's Licencing
  terms.
* The Sun version does not need to be Registered with Netscape, so the
  Help=>Registration_Information choice was removed.
* The Release Notes and Support items point to Sun Service's web server
  http://access1.sun.com/.

1.0 to 1.12 Changes

* New Icon and Animation: The throbbing N has breathed its last. Christopher Skinner is the winner of the No Throbbing, Breathing, Pulsing N contest. For more information on Chris and other Netscape contest winners, see "We have winners" at the URL http://home.netscape.com/home/contest. Click on the new icon for a short cut to Netscape's home page.

* Remote Control Mechanism: You can control Netscape from outside of the application using X properties. For more information, see "Netscape Client APIs" at the URL http://home.netscape.com/newsref/std/index.html#client_api. .

* Cache Preferences: There are several Preferences available that refer to the Memory cache and the Disk cache. You can set the size of these caches through the Preferences window. A larger Disk cache means that as you browse, files will be saved on your disk so that when you re-visit them, they are brought up from disk rather than from the network.

  In Netscape 1.0, when you asked for a document already in your cache, Netscape would always contact the original document server and ask for the document's header information, for comparison. This affected the

  performance of the cache. In 1.12, you can control this through the Cache preference: Verify Documents. If you set this to "Every Time", you will have the same behavior as in Netscape 1.0. "Once per session" means that the document is only checked the first time you use it in a particular Netscape session. "Never" means that the server is contacted only if the document's header information indicates that it has expired. For the best cache performance, use "Once per session".

* Security: The US version of the software now supports numerous cipher choices: RC4-40, RC4-128, RC2-40, RC2-128, DES-64, and DES-192. The cipher used in any particular document is displayed in the File Menu/Document Information dialog box. The export version of the software, including version 1.0, supports only RC4-40.

* Secure News: There is support for secure news servers. The url for secure news is "snews://". You can see an example of secure news in our Netscape User's Group at the URL

  http://home.netscape.com/commun/netscape_user_groups.html.

* News and Mail Improvements: Improvements to the News and Mail

---

interface, including:
  o The Newsgroup button has returned.
  o Subscribing to groups on multiple news hosts is supported.
  o A new Mail Message window that also includes News Posting
    capabilities and file attachments. Binary files that are mailed
    or posted use base64 encoding. Attached files with high bit
    characters or long lines are encoded using quoted-printable.
  o Improved presentation of newsgroups and articles. In addition,
    quoted-printable and base64 articles and attachments are decoded
    correctly.
  o A new Preference to set the maximum number of articles to
    download and thread at a time: a low number will give much better
    performance over a modem line.
  o A new Preference to allow 8-bit or MIME Compliant
    Quoted-Printable Content-Transfer-Encoding for mail and news
    articles that contain high bit characters. The default is to
    "Allow 8-bit".
  o MIME-compliant news articles will be presented according to their
    content type. (This means that news articles may contain HTML, or
    even encoded JPEG or GIF images. Messages of type multipart/mixed
    are also understood.)
  o Signature files are now supported in mail and in news.

* Dynamic Documents: You can create documents whose contents are updated
  dynamically, either on a periodic basis or irregularly. See the URL

  http://home.netscape.com/assist/net_sites/dynamic_docs.html for
  details.

* Performance: Many operations should be faster. Going back to visited
  pages is now faster, because we've devoted 2/3 of the memory cache to
  decoded images.

  Startup time should be improved.

* Tables: These are as defined in the current HTML 3.0 draft
  specification at the URL
  http://www.w3.org/hypertext/WWW/Markup/Markup.html . For a
  demonstration of tables, see the URL
  http://home.netscape.com/assist/net_sites/tables.html. We've also
  improved the appearance of table bevels when they're over a
  background.

* Backgrounds: ... and the ability to change text colors. For a
  demonstration, see the URL
  http://home.netscape.com/assist/net_sites/bg/index.html. The
  demonstration pages will show you the correct HTML tag.

* User Defined Backgrounds: You can now specify a default background
  image, as well as background colors in the Preferences on the Mac and
  in Windows. You can specify them in the defaults file in Unix.

* ALT tags & Loading Images: We now show ALT text in images and allow
  you to load individual images when Auto Load Images is turned off. To
  load an image, click on the image replacement icon. To follow the
  image's link, click on the ALT text, or use the popup menu.

* Save Next Link as... and more in a Popup Menu: A Popup menu with some
  key commands right at your fingertips. Position the mouse in the
  document area. On Windows and Unix, hold the right mouse button down.
  On the Macintosh, hold the mouse button down for about 1 second. A
  popup menu will appear. Different menu commands are activated
  depending on whether the mouse is over an image, a link, or document
  whitespace.

* Disabled links during Save As...: When Netscape is saving a file to
  disk, a busy cursor appears in the window from which the download was
  initiated. You will not be able to click on links in this window until
  the download is complete. This is to make it harder to accidentally
  abort the transfer by clicking on another link in the document area.
  You can open another window and do work there while you are waiting
  for the transfer to complete. Be careful: the menu items and buttons
  in the downloading window are still operable while Netscape is saving
  the file. If you go back to the downloading window and hit the stop
  button, or click on a button or menu item that takes you to a new URL,
  you will cancel the download.

* Preference window: The preference window has been slightly rearranged.

* -iconic: The -iconic command line option works again. For a list of
  command line options, invoke netscape -help.

* Control over Color Usage: in addition to the installColormap resource,
  which causes Netscape to install a private colormap, Netscape 1.12 has
  a maxImageColors resource and corresponding -ncols command-line option
  which will limit the number of colors which Netscape will allocate

from the default colormap.

* GrayScale: The GrayScale visual is now supported.

* External Viewer errors: Should an external viewer produce output to
  stdout or stderr, that information will be presented in a dialog box
  instead of being silently discarded.

* Compression and External Viewers: Documents encoded with `compress' or
  `gzip' will be correctly decoded before being displayed or handed to
  an external viewer.

* Selection & Dragging: Dragging the selection with the left mouse
  button will now scroll the document as necessary.

* Bug fixes: Many fixes since 1.0.

   o Redisplay Bug: We have made some code changes to fix a bug where
     the display was not being cleared before the new page was drawn.
     We have also discovered some redisplay problems with BackingStore
     turned on. Go to the Known Problems/Bugs section of these release
     notes for more information.

   o Cache Bug: A bug in the disk cache was fixed which would
     sometimes cause Netscape to present out-of-date documents.

     IMPORTANT: After upgrading to 1.12 it is recommended that you
     flush your disk cache to remove any incorrect information that
     may have been saved by an earlier release of the Netscape
     Navigator.

   o Support for pending certificates has been added. Servers can swap
     between old and new certificates without generating error
     messages on the client.

UPGRADING FROM 1.0 OR 1.1

* Upgrading from Version 1.0 or earlier: If you are upgrading directly

  to Netscape Navigator 1.12 from Version 1.0 or 1.0N, or one of the
  earlier betas, you need to upgrade your $HOME/.MCOM-* configuration
  files by hand.

While running the Version 1.0 Navigator, change your cache directory setting in preferences from $HOME/.MCOM-cache to $HOME/.netscape-cache. Exit the Navigator and delete your old $HOME/.MCOM-cache directory. You can then rename the rest of the configuration files:

```
$HOME/.MCOM-prefereces      -> $HOME/.netscape-preferences
$HOME/.MCOM-bookmarks.html  -> $HOME/.netscape-bookmarks.html
$HOME/.MCOM-global-history  -> $HOME/.netscape-history
$HOME/.MCOM-HTTP-cookie-file -> $HOME/.netscape-cookies
```

* Upgrading from Version 1.1: If you are upgrading to version 1.12 from version 1.1, you do not have to do anything to your Netscape dot files.

KNOWN PROBLEMS/BUGS

* Packages Required to Use Localized Navigator:If you get the following error message, you may be missing some packages:

```
ld.so.1: netscape:fatal: /usr/openwinlib/locale/fr/XLIBI18N/xlibil8n.so.1:
can't open file: err no 22
netscape: locale 'fr' not supported by xlib; trying 'C'
```

If you get this error you need to make sure the following packages are installed:

o SUNWxil8n

o SUNWplow

o SUNWplow1

* Suggested .Xdefaults entry for the "ja" locale: A better font for the "ja" locale is:

```
*fontList:-sun-gothic-medium-r-normal--16-140-75-75-c-70-\
jisx0201.1976-0;\
-sun-gothic-medium-r-normal--16-140-75-75-c-140-\
jisx0208.1983-0:
```

NOTE: Other locales may need to come up with their own customized

fonts.

* Motif Runtime Libraries Required for Solaris 2.3 Machines: The 1.12SUN
  Navigator is dynamically linked with the X and Motif libraries. If the
  Mofit dynamic libraries are not properly installed into /usr/dt/lib,
  the Navigator will exit with this error:

    ld.so.1: netscape: fatal: libXm.so.3:can't open file: errno=2

  The Motif runtime libraries are in the SUNWmfrun package. It can be
  found on the Solaris 2.4 SDK or the Solaris 2.4 OS CD.

  NOTE: The SUNWmfrun package was designed to correctly install and run
  on Solaris 2.3.

* Convert your Version 1.0 dot files before running Version 1.12: If you
  still have Version 1.0 configuration files in your home directory when
  you start version 1.12, it will segmentation fault on startup. You
  need to convert your $HOME/.MCOM-* files before running the version
  1.12 Navigator. See the "Upgrading from 1.0 or 1.1" section for
  instructions on how to do this.

* Newsrc file not updating: If your $TMPDIR environment variable is set
  to a directory that is on a different file system than your .newsrc
  directory, Netscape will not be able to save changes to your .newsrc
  file(s). It will not update the article listings or the number of
  unread articles.

  There are two work arounds. You can change your $TMPDIR environment
  variable so that it uses a directory on the same file system as your
  .newsrc directory (Check Options Menu - Preferences - Mail and News
  for the location of your News RC Directory). If your $TMPDIR variable
  can't be changed because it is needed by other applications, you can
  run Netscape from within a shell script that re-sets the $TMPDIR
  environment variable for that shell only.

* Newsrc file damaged or not updating large newsgroups: The .newsrc
  entry for each newsgroup has a line limit of 1024 bytes. If you have
  read a lot of articles in a newsgroup, your .newsrc entry may be
  getting truncated. You can hit the Mark All Articles Read button to
  compress the entry.

* Redisplay problems with BackingStore: Netscape will not redisplay
  properly when scrolling if BackingStore has been turned on for the

Netscape windows. Netscape attempts to turn off BackingStore on its own windows, but does not succeed in doing so if you have started your X server with the argument which tells it to always use BackingStore. On many X servers, this is the -wm flag. A workaround to this problem is to not pass the -wm flag to your X server.

* Redisplay problems with Sun TCX 24-bit Frame Buffer: Netscape will not

  redisplay properly when using the Sun TCX 24-bit Frame Buffer on Solaris 2.4 without Sun Patch 101923-06 or later. The symptoms are similar to the BackingStore problem (above).

* Redisplay problems after changing Toolbar: Text and backgrounds may fail to redraw during scrolling if you have recently changed the Window Styles Preference to "Show Toolbar As: Pictures". To get rid of this problem, create a new window, then close the old one.

* Mixed colors with different endian-ness: When Netscape is running on one machine and displaying on another in TrueColor mode, and those two machines have different endian-ness (bit-order) the red and green colors will be reversed. The only workaround is to use PseudoColor mode (8-bit, invoked with the netscape -visual pseudocolor command-line option) or to run Netscape on a machine with the same bit-order as the machine on which it is being displayed.

* Searching tables: You can't search for text that is contained within a table.

* Selections in tables: In rare cases, selecting text across multiple cells of a table can cause Netscape to go into an infinite loop. Try to avoid selecting multiple table cells in nested tables.

* Connecting to U.S. servers with Export clients: The export version of Netscape 1.12 supports only the RC4-40 cipher. Using the export version to connect to servers that support other ciphers should give you an error message saying that the cipher types don't match. On OSF1, you will get a -12286 TCP error saying the server may be down or unreachable. On Linux and Solaris you do not get a message at all.

* Obsolete X Resources: Some people have reported two Save buttons on the View Source window, and other odd problems. This occurs because there is an obsolete copy of the Netscape.ad file installed. Check for a file named "Netscape" in your home directory.

Instead of installing the whole Netscape.ad file, we recommend that
you copy the few resources you need and put them in your .Xdefaults
file in your home directory. It's too easy to get the resources mixed
up between versions if you install the whole file.

```
----------------------------------------------------------------------
         Copyright ⌐ 1995 Netscape Communications Corporation
              Copyright ⌐ 1995 Sun Microsystems, Inc
----------------------------------------------------------------------
```

Copyright 1995 Sun Microsystems, Inc., 2550 Garcia Avenue, Mt. View, CA

94043-1100 USA. All rights reserved.

This page intentionally left blank